



Universidad
Carlos III de Madrid

Departamento de Telemática

PROYECTO FIN DE CARRERA

Grado en Ingeniería de Sistemas de Comunicaciones

Desarrollo de una aplicación para la gestión de PFCs/TFGs para dispositivos móviles Android

Autor: Eduardo P. García del Valle

Tutor: Florina Almenares Mendoza

Leganés, Marzo de 2014

Título: Desarrollo de una aplicación para la gestión de PFCs/TFGs para dispositivos móviles Android

Autor: Eduardo P. García del Valle

Director: Florina Almenares Mendoza

EL TRIBUNAL

Presidente: _____

Vocal: _____

Secretario: _____

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día __ de
_____ de 20__ en Leganés, en la Escuela Politécnica Superior de la Universidad
Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos

Quiero agradecer a mi familia, a mis amigos y a mis profesores por su apoyo y ayuda para realizar este Trabajo de Fin de Grado

Además, quiero expresar mi más sincero agradecimiento a mi tutora Florina, a mis profesores y a los miembros del Departamento de Servicios Informáticos de la UC3M, sin cuyo apoyo y ayuda no hubiese podido realizar este Trabajo de Fin de Grado.

Resumen

La realización del PFC/TFG es el último reto al que se enfrentan los estudiantes universitarios antes de finalizar sus estudios en la Universidad Carlos III. El acceso a la información sobre la oferta existente, las gestiones burocráticas necesarias y el control del material relacionado se realizan actualmente mediante diferentes herramientas de forma que no existe una aplicación que centralice todas las tareas y facilite la labor de los estudiantes.

Observando esta situación surgió la idea de desarrollar un sistema de búsqueda y gestión de proyectos de fin de grado para dispositivos móviles integrado con las aplicaciones existentes mediante servicios web. El propósito final del proyecto es proporcionar a alumnos y profesores una herramienta ágil que integre todas las funciones necesarias para la selección y el desarrollo de PFC/TFG.

Para la realización del proyecto se ha llevado a cabo un análisis exhaustivo de las tecnologías disponibles para su implementación, así como de otras aplicaciones similares y sus casos de uso. A lo largo de este documento se revisarán los objetivos que se han fijado para el sistema, la forma en que se ha diseñado e implementado y las pruebas y conclusiones que se han obtenido.

La aplicación está principalmente orientada a alumnos, si bien se pretende que permita a los tutores realizar un mejor seguimiento de la evolución de los proyectos que dirigen. En el diseño e implementación de la aplicación ha tenido en cuenta especialmente la adaptabilidad y modularidad de las funcionalidades, de forma que en el futuro otros estudiantes puedan mantener, extender y reutilizar la totalidad o partes de la aplicación.

La aplicación de gestión de proyectos de fin de grado es la primera que utiliza la nueva API JSON para servicios de la Universidad Carlos III. Por tanto, otro de los objetivos cumplidos con la aplicación es la de depurar y mejorar el funcionamiento de dichos servicios, así como sentar las bases para futuras aplicaciones basadas en ellos.

Palabras clave: Android, Servicios Web, JSON, trabajo de fin de grado, proyecto de fin de carrera, búsqueda avanzada, seguimiento, cliente, servidor, aplicaciones móviles.

Abstract

Writing the Final Degree Project is the last challenge that students must face before finishing their careers at Universidad Carlos III. The access to the information about the available options, the required bureaucracy or the management of the related material are currently performed through different tools, since there isn't any application that centralizes all these tasks and makes the work of the students easier.

Observing this situation, came the idea of making an application to find and manage Final Degree Projects which is connected to the existing applications via web services. The final purpose of this application is to provide students and tutors with a convenient tool that integrates all the necessary features for the selection and development of Final Degree Projects.

For the development of this project, an exhaustive analysis of the currently available technologies for its implementation as well as of other similar applications and use case has been carried out. Throughout this document, the requirements of the system, its design and implementation and finally the test and the obtained conclusions will be reviewed,

The application is mainly oriented to students, though it's also aimed that tutors may improve the tracking of the projects they are leading. For the design and implementation of the application, the adaptability and modularity of its features has been seriously taken into account, so that future students may maintain, extend and totally or partially reuse it.

This is the first application based on the new JSON API for services of Universidad Carlos III. Thus, other of the fulfilled goals of the application has been the debugging and improvement of these services, and also to establish a reference for future applications based on the same or similar services.

Keywords: Android, Web Services, JSON, Final Degree Projects, Advanced search, Tracking, Client, Server, Mobile Application.

Índice General

1.	Introducción y objetivos	17
1.1.	Introducción.....	18
1.2.	Motivación.....	18
1.3.	Objetivos.....	18
1.4.	Fases del desarrollo.....	19
1.5.	Medios empleados	19
1.6.	Acrónimos y abreviaturas	20
1.7.	Contenido de la memoria.....	21
2.	Estado del Arte	23
2.1.	Introducción.....	24
2.2.	Aplicaciones móviles de la UC3M	24
2.3.	Servicios y aplicaciones web de la UC3M	27
2.4.	Sistemas operativos para dispositivos móviles	29
2.5.	Sistema operativo Android	30
2.6.	Servicios Web.....	38
2.7.	Marco regulatorio	41
3.	Análisis del sistema	43
3.1.	Introducción.....	44
3.2.	Descripción general	44
3.3.	Requisitos de usuario.....	46
3.4.	Requisitos de software.....	51
4.	Diseño	56
4.1.	Arquitectura del sistema	57
4.2.	Diseño de la aplicación cliente	60
4.3.	Diseño de la aplicación servidora	75
5.	Implementación	76
5.1.	Modelo de datos interno de la aplicación	77
5.2.	Integración con el servicio web JSON.....	77
5.3.	Obtención de datos de páginas web.....	79

5.4.	Implementación de las interfaces gráficas de usuario	81
5.5.	Integración con otras aplicaciones.....	83
6.	Pruebas.....	84
6.1.	Introducción.....	85
6.2.	Entornos de prueba	85
6.3.	Cumplimiento de los requisitos de usuario.....	86
7.	Planificación y presupuesto	88
7.1.	Planificación	89
7.2.	Presupuesto	89
8.	Conclusiones.....	91
8.1.	Conclusiones.....	92
8.2.	Líneas futuras	92
	Referencias.....	94
ANEXO A.	Guía de usuario	98
ANEXO B.	Guía de implantación	111
ANEXO C.	Material entregado	113

Índice de Tablas

Tabla 1 - Comparativa de características en los principales SO para móviles.....	31
Tabla 2 - Plantilla de requisito de usuario.....	46
Tabla 3: RUC-01	46
Tabla 4: RUC-02	47
Tabla 5: RUC-03	47
Tabla 6: RUC-04	47
Tabla 7: RUC-05	47
Tabla 8: RUC-06	47
Tabla 9: RUC-07	48
Tabla 10: RUC-08	48
Tabla 11: RUC-09	48
Tabla 12: RUC-10	48
Tabla 13: RUC-11	48
Tabla 14: RUC-12	49
Tabla 15: RUC-13	49
Tabla 16: RUC-14	49
Tabla 17: RUC-15	49
Tabla 18: RUC-16	49
Tabla 19: RUR-01	50
Tabla 20: RUR-02	50
Tabla 21: RUR-03	50
Tabla 22: RUR-04	50
Tabla 23: RUR-05	50
Tabla 24: RUR-06	51
Tabla 25: RUR-07	51
Tabla 26: RUR-08	51
Tabla 27: RSF-01	52
Tabla 28: RSF-02	52

Tabla 29: RSF-03	52
Tabla 30: RSF-04	52
Tabla 31: RSF-05	52
Tabla 32: RSF-06	53
Tabla 33: RSF-07	53
Tabla 34: RSF-08	53
Tabla 35: RSF-09	53
Tabla 36: RSF-10	53
Tabla 37: RSF-11	54
Tabla 38: RSF-12	54
Tabla 39: RSF-13	54
Tabla 40: RSNF-01	54
Tabla 41: RSNF-02	54
Tabla 42: RSNF-03	55
Tabla 43: RSNF-04	55
Tabla 44: RSNF-05	55
Tabla 45: RSNF-06	55
Tabla 46: RSNF-07	55
Tabla 47 – Registro de proyectos recientes	74
Tabla 48 – Registro de proyectos favoritos	74
Tabla 49 – Registro de perfil de usuario	74
Tabla 50 – Características técnicas del emulador para las pruebas	85
Tabla 51 – Características técnicas del emulador para las pruebas	85
Tabla 52 – Evaluación del cumplimiento de los requisitos de capacidad	86
Tabla 53 – Evaluación del cumplimiento de los requisitos de restricción	87
Tabla 54 – Planificación del proyecto	89
Tabla 55 – Costes de personal	89
Tabla 56 – Costes de hardware	90
Tabla 57 – Costes de hardware	90
Tabla 58 – Coste total del proyecto	90

Índice de Figuras

Figura 1 – Aplicaciones de la UC3M para iOS (1)	24
Figura 2 – Aplicaciones de la UC3M para Android (2)	25
Figura 3 – Vista principal de la aplicación UC3M.....	25
Figura 4 – Vista de la aplicación Mapas	26
Figura 5 – Vista de la aplicación Agenda.....	26
Figura 6 – Vista de la aplicación Calificaciones	26
Figura 7 – Vista de la aplicación PAU	26
Figura 8 – Tablón de proyectos de fin de carrera y fin de grado (3).....	28
Figura 9 – Listado de proyectos de un departamento.....	28
Figura 10 - Evolución de las ventas de <i>smartphones</i> por sistema operativo (10).....	31
Figura 11 – Distribución del porcentaje de dispositivos Android por versión del OS (16)	33
Figura 12 - Arquitectura del Sistema Operativo Android (19)	35
Figura 13 – Comparación de una respuesta JSON (izquierda) y XML (derecha) con la misma información (28)	40
Figura 14 - Arquitectura Cliente-Servidor (30).....	57
Figura 15 - Arquitectura MVC aplicada al diseño de aplicaciones Android (31).....	58
Figura 16 - Arquitectura completa del sistema	59
Figura 17 - Vista de escritorio	60
Figura 18 - Vista de acceso	60
Figura 19 - Vista de menú principal.....	60
Figura 20 - Vista de tablón	61
Figura 21 - Vista de listado de proyectos	61
Figura 22 - Vista de detalle de proyecto.....	61
Figura 23 - Vista de tutor	62
Figura 24 - Vista de contacto	62
Figura 25 - Vista de buscador.....	62
Figura 26 - Vista de información	63
Figura 27 - Vista de detalle de información	63

Figura 28 - Vista de perfil	63
Figura 29 - Vista de matrícula.....	64
Figura 30 - Vista de proyecto	64
Figura 31 - Vista de seguimiento	64
Figura 32 - Vista de agenda.....	65
Figura 33 - Vista de añadir/editar evento	65
Figura 34 - Vista de documentos.....	65
Figura 35 - Vista de documentos.....	65
Figura 36 - Vista de acerca de	65
Figura 37 – Diagrama de navegación.....	66
Figura 38 - Diagrama funcional de las clases de la aplicación	68
Figura 39 - Diagrama UML de clases de control de vistas (Actividades)	68
Figura 40 - Diagrama UML de clases de control de vistas (Fragmentos).....	69
Figura 41 - Diagrama UML de clases entidades y adaptadores	70
Figura 42 - Diagrama UML de clases de acceso al modelo local	71
Figura 43 - Diagrama UML de clases de acceso al modelo remoto.....	72
Figura 44 - Diagrama UML de clases de utilidad	73
Figura 45 - Modelo de ejecución asíncrono de tareas (38)	78
Figura 46 – Enlaces con información de PFCs y TFGs	80
Figura 47 – Versiones inicial y final del estilo de la aplicación.....	81
Figura 48 – Guía de usuario: icono de la aplicación	99
Figura 49 – Guía de usuario: pantalla inicial	100
Figura 50 – Guía de usuario: menú de navegación	100
Figura 51 – Guía de usuario: tablón proyectos por departamentos y titulaciones ...	101
Figura 52 – Guía de usuario: buscador.....	102
Figura 53 – Guía de usuario: selección de departamento en el buscador.....	103
Figura 54 – Guía de usuario: selección de palabras clave en el buscador.....	103
Figura 55 – Guía de usuario: listado de proyectos	104
Figura 56 – Guía de usuario: detalles de proyecto	105
Figura 57 – Guía de usuario: tutor del proyecto.....	106
Figura 58 – Guía de usuario: alumno del proyecto	106
Figura 59 – Guía de usuario: acta del proyecto.....	107

Figura 60 – Guía de usuario: mi proyecto	107
Figura 61 – Guía de usuario: opciones del perfil de usuario.....	108
Figura 62 – Guía de usuario: información sobre proyectos	108
Figura 63 – Guía de usuario: ayuda.....	109
Figura 64 – Guía de usuario: acerca de	110
Figura 65 – Guía de usuario: créditos	110

1. Introducción y objetivos

1.1. Introducción

La finalidad de este capítulo es dar a conocer el contenido de este documento y que sirva como un primer encuentro con el proyecto de final de carrera que se ha realizado. Este capítulo sirve además como una pequeña introducción de los conceptos que se explican en los sucesivos apartados del documento.

1.2. Motivación

Desbordando las previsiones sobre la computación ubicua, en los últimos 10 años el auge de las aplicaciones para dispositivos móviles ha obligado a los proveedores de servicios a tener en cuenta estas plataformas a la hora de desarrollar y distribuir sus productos.

Actualmente la Universidad Carlos III de Madrid ofrece a través de su web acceso a distintos servicios, si bien no se ha optimizado su acceso desde dispositivos móviles así como su integración con otras aplicaciones. Poco a poco se han ido creando aplicaciones móviles que se encargan de cubrir este vacío para distintos servicios (Agenda, Información, Mapas...).

El sistema que se desea construir permitirá la gestión de todas las tareas relacionadas con el proyecto de fin de carrera, desde su búsqueda, selección, desarrollo, seguimiento y presentación. La aplicación está en principio orientada a su uso por estudiantes, si bien se contempla su extensión para añadir funcionalidades específicas para tutores. Este nuevo sistema no pretende la sustitución de la aplicación web actual, sino la integración con la misma para ofrecer funcionalidades comunes y complementarias aprovechando las posibilidades que ofrecen las aplicaciones móviles.

1.3. Objetivos

El objetivo de este proyecto es desarrollar un sistema de búsqueda y gestión de PFCs y TFGs para dispositivos móviles integrado con un servidor de información mediante servicios web, proporcionando una herramienta ágil que integre todas las funciones necesarias para la selección y el desarrollo de proyectos de fin de grado.

Específicamente, este proyecto abarcará únicamente la aplicación de búsqueda y gestión para alumnos, dejando como trabajo futuro el desarrollo de la herramienta de gestión tipo *back-office* para profesores.

Este objetivo se engloba dentro de la mejora de los servicios de la Universidad Carlos III proporcionando acceso a las funciones relacionadas con Trabajos de Fin de Grado desde dispositivos móviles.

1.4. Fases del desarrollo

El desarrollo de este proyecto se ha realizado en varias fases. La primera de ellas consistió en la evaluación de la viabilidad con los responsables de las aplicaciones de la UC3M.

En una segunda fase, se elaboró un documento de Especificación de Requisitos de Software donde se detallaban las funcionalidades previstas para la aplicación, conforme a las condiciones indicadas.

A continuación se inició la fase de desarrollo propiamente dicha, a lo largo de la cual y de forma iterativa se fueron implementando distintas partes de la aplicación y la memoria, se revisaban junto a la tutora en reuniones periódicas, y se corregían los aspectos acordados.

Una vez finalizada la primera versión de la aplicación se mantuvo una reunión con los responsables de las aplicaciones de la UC3M para su evaluación preliminar. Como parte de esta reunión se trató la futura evolución de la aplicación del tablón con vistas a la adaptación completa a grados, y se propuso realizar dos versiones de la aplicación para móviles: una que incluyese la funcionalidad básica (principalmente búsqueda) y otra que adelantase algunas de las futuras funcionalidades (perfil de estudiante, búsqueda avanzada...).

En una fase final se realizaron ligeros ajustes técnicos en la aplicación y en el texto de la memoria, como paso previo a su entrega y defensa.

1.5. Medios empleados

Para la realización de este proyecto se han empleado las siguientes herramientas de hardware y software.

1.5.1. Hardware

El hardware necesario ha incluido:

- Entorno de desarrollo: Ordenador portátil HP ProBook 4520s
- Entornos de pruebas: Tablet Samsung Galaxy Note 10.1 y teléfono móvil Samsung Galaxy S3

1.5.2. Software

El software utilizado es el siguiente:

- Sistema operativo: Windows 7 Professional 64 bits

- Herramienta de desarrollo: Eclipse Índigo SR2 (1) con Android Development Tools (Rev. 22)
- Plataforma de desarrollo: Java JDK 6 y Android ADK (API 19)
- Microsoft Word 2010 (Evaluación)

1.6. Acrónimos y abreviaturas

A continuación se definen los acrónimos empleados en este documento:

- ADK *Android Development Kit*
- AJAX *Asynchronous Javascript And XML*
- API *Application Programming Interface*
- App *Application*
- AVD *Android Virtual Device*
- BSD *Berkeley Software Distribution*
- CRUD *Create, Read, Update and Delete*
- GNU *GNU is Not Unix* (Acrónimo recursivo)
- GUI *Graphical User Interface*
- HTTP *Hypertext Transfer Protocol*
- IDE *Integrated Development Environment*
- JMS *Java Message Service*
- JSON *JavaScript Object Notation*
- JDK *Java Development Kit*
- MVC *Model-View-Controller*
- OASIS *Organization for the Advancement of Structured Information Standards*
- OS *Operating System*
- PFC Proyecto de Fin de Grado
- PIN *Personal Identification Number*
- SMTP *Simple Mail Transfer Protocol*
- SOAP *Simple Object Access Protocol*
- SQL *Structured Query Language*
- TCP *Transmission Control Protocol*
- TFG Trabajo de Fin de Grado
- UC3M Universidad Carlos III de Madrid
- UML *Unified Modeling Language*

- URI *Uniform Resource Identifier*
- URL *Uniform Resource Locator*
- W3C *World Wide Web Consortium*
- WSDL *Web Services Description Language*
- XML *Extensible Markup Language*

1.7. Contenido de la memoria

En este apartado se va a proceder a explicar el contenido de este documento, viendo cada uno de los capítulos que nos podemos encontrar, los cuales componen este documento.

- **Capítulo 2. Estado del arte**, se mostrará una visión de las tecnologías y aplicaciones disponibles para realizar el proyecto.
- **Capítulo 3. Análisis del sistema**, se definen los requisitos de la aplicación analizando todas las funcionalidades que debe cumplir.
- **Capítulo 4. Diseño**, en base a los requisitos planteados y a las tecnologías disponibles, se describe el diseño para la aplicación.
- **Capítulo 5. Implementación**, se describen los diferentes detalles de cada uno de los módulos que contiene el sistema desde el punto de vista de la implementación, así como las diferentes estrategias seguidas a la hora de implementar el sistema.
- **Capítulo 6. Pruebas**, se detallan las pruebas llevadas a cabo para cada una de los requisitos definidos durante el análisis y se evalúa su grado de cumplimiento.
- **Capítulo 7. Planificación y presupuesto**, contiene la planificación y el presupuesto necesarios para realizar el proyecto.
- **Capítulo 8. Conclusiones**, se indican los aspectos positivos y negativos que se extraen tras la realización del proyecto y se plantean algunas líneas de mejora para el futuro.
- **Referencias**: Se enumera todo el material consultado a la hora de realizar el presente documento.

Además se incluyen una serie de anexos para la correcta utilización y comprensión del sistema.

- **Anexo A. Manual de usuario**, el cual sirve de guía de manejo de la aplicación a nivel de usuario, sin entrar en detalles técnicos.

- **Anexo B. Manual de implantación,** contiene la información necesaria para instalar la aplicación, así como los requisitos de software y hardware.
- **Anexo C. Material entregado,** registro de todo el material entregado como parte del proyecto.

2. Estado del Arte

2.1. Introducción

En este capítulo se presenta un estudio de las diferentes tecnologías disponibles para implementar el sistema descrito en el capítulo anterior. De esta forma se realiza un estudio de las tecnologías que hay hasta el momento, con el fin de obtener un producto final de calidad.

Además en este capítulo se van a analizar productos similares que existen en el mercado comentando sus características, y finalmente realizando una comparativa entre todos ellos, analizando las ventajas que aportan cada uno de ellos en función con el sistema a desarrollar en este producto.

El objetivo final de este estudio es obtener una mayor visión del entorno tecnológico y así valorar otros productos similares evaluando si el producto desarrollado es viable en la actualidad.

2.2. Aplicaciones móviles de la UC3M

La UC3M distribuye actualmente un conjunto de aplicaciones para dispositivos móviles y tabletas en versiones para iOS y Android. El objetivo de estas aplicaciones es ofrecer servicios de la Universidad, algunos ofrecidos ya vía Web, adaptados, extendidos y/o optimizados para este tipo de terminales.



Figura 1 – Aplicaciones de la UC3M para iOS (2)



Figura 2 – Aplicaciones de la UC3M para Android (3)

Las aplicaciones ofertadas actualmente son:

- **UC3M:** Esta aplicación actúa realmente como un agregador de servicios, entre los que se encuentran los estudios ofertados, el archivo multimedia, el directorio corporativo y la localización, tal como muestra la Figura 3. Se distribuye tanto para iOS como para Android.



Figura 3 – Vista principal de la aplicación UC3M

- **Mapas:** Localiza en un mapa las ubicaciones de la UC3M mediante el código de edificio o despacho (**¡Error! No se encuentra el origen de la referencia.**). Disponible para iOS y Android.
- **Agenda:** Contiene el listado y calendario de eventos de la UC3M (**¡Error! No se encuentra el origen de la referencia.**). Se encuentra disponible para iOS y Android.
- **Calificaciones:** Permite a los alumnos de la UC3M acceder a sus

calificaciones. Para ello deben autenticarse con su usuario y contraseña. Sólo se encuentra disponible para Android (Figura 6).

- **PAU:** La aplicación permite consultar la calificación en la Prueba de Acceso a la Universidad. Sólo está disponible para Android (Figura 7).

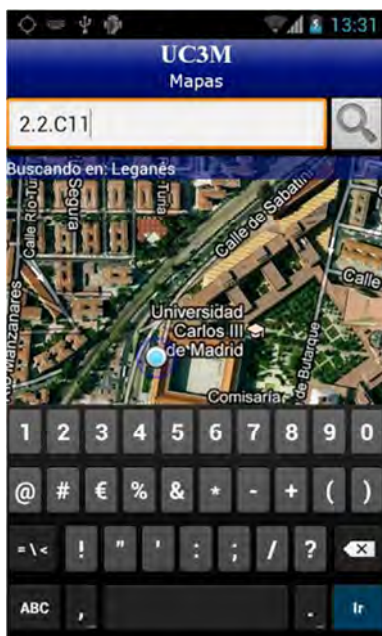


Figura 4 – Vista de la aplicación Mapas



Figura 5 – Vista de la aplicación Agenda



Figura 6 – Vista de la aplicación Calificaciones



Figura 7 – Vista de la aplicación PAU

2.3. Servicios y aplicaciones web de la UC3M

La UC3M ofrece a través de su web un conjunto de servicios para facilitar las operaciones que habitualmente realizan alumnos y profesores. Estos servicios están agrupados en aplicaciones. En este apartado se estudian sólo las que se han considerado relevantes para el desarrollo del proyecto:

2.3.1. *Campus Global*

Es la principal aplicación web de la UC3M y agrega o da acceso a la mayoría de los servicios ofrecidos por la universidad. Un mismo servicio puede contener distintas opciones en función del usuario (alumno o profesor, por ejemplo):

- **Aula Global 2:** Es una aplicación *Moodle* integrada por la UC3M para la gestión de recursos docentes para alumnos y profesores.
- **Secretaría:** Permite realizar a distancia las operaciones de secretaría más habituales, como la matriculación, la solicitud de dispensa o la consulta de calificaciones.
- **UC3M en la nube:** Se basa en aplicaciones de Google (Gmail, Drive, Calendar) para proporcionar servicios avanzados de correo, almacenamiento y agenda, entre otros.

2.3.2. *Aplicación de búsqueda de PFCs y TFGs*

Perspectiva para alumnos

La UC3M dispone de una aplicación web para la búsqueda de PFCs y TFGs de forma centralizada en la dirección <http://www3.uc3m.es/web/pfc/index.xml>. Hasta su aparición, cada departamento (e incluso cada profesor) anunciaba de forma separada la oferta de proyectos disponibles.

Esta es la aplicación que se ha tomado como referencia para desarrollar la aplicación de búsqueda y gestión de PFCs y TFGs. La Figura 8 y la Figura 9 muestran, respectivamente, el tablón de departamentos y el listado de proyectos. El objetivo es que la aplicación móvil mantenga en la medida de lo posible la misma estructura.

Por ejemplo, en la Figura 9 se muestra como el estado del TFG/PFC se identifica mediante un código de color:

- Libre u ofertado (Verde)
- Adjudicado (Amarillo)
- Pendiente de lectura (Rojo)
- Leído (Rojo)

Este criterio se mantendrá también en la aplicación cliente.

<div>  <div> <div>Universidad Carlos III de Madrid</div> <div>Proyectos Fin de Carrera y Trabajos Fin de Grado</div> </div> </div>				
Oferta de proyectos Fin de Carrera y Trabajos Fin de Grado				
<p>La matrícula y lectura del proyecto fin de carrera para el curso académico 2009/10 comprende desde el 01 de enero al 31 de octubre del 2010</p> <p>Haga click sobre un departamento para ver la oferta de proyectos y trabajos del mismo.</p>				
Departamento	Libres	Adjudicados	Pendientes de lectura	TOTAL
Bioingeniería y Aeroespacial	17	14	0	31
CC e Ing.Materiales e Ing.Quim	32	35	2	69
Estadística	8	3	0	11
Física	13	9	0	22
Informática	314	279	24	617
Sistemas y Automática	84	67	3	154
Ingeniería Eléctrica	7	99	0	106
Ingeniería Mecánica	72	177	6	255
Ingeniería Telemática	174	103	4	281
Ingeniería Térmica y Fluidos	34	63	4	101
Matemáticas	9	2	0	11
Mec. Med. Cont. y Teor. Estruct.	30	35	2	67
Tecnología Electrónica	49	100	4	153
Teoría de la Señal y Comunicac	75	55	0	130
	918	1041	49	2008

Figura 8 – Tablón de proyectos de fin de carrera y fin de grado (4)

Oferta de proyectos Fin de Carrera y Trabajos Fin de Grado					
DEPARTAMENTO DE INGENIERÍA TELEMÁTICA					
Aqui tiene la oferta de Proyectos Fin de Carrera y Trabajos Fin de Grado del DEPARTAMENTO DE INGENIERÍA TELEMÁTICA. Puede ordenar las filas eligiendo la columna que desee.					
FILTRO ► Mostrar sólo proyectos o trabajos que contengan las palabras					
TÍTULO Y DESCRIPCIÓN	REQUISITOS	RESPONSABLES	CONTACTO	OFERTADO	ESTADO
ESTUDIO DE MODELOS DE CRÉDITOS EN OVERLAY NETWORKS Este proyecto se enmarca dentro de los objetivos del proyecto europeo CONTENT (Content Networks and Services for Home Users). Uno de los objetivos de las Community Networks es ofrecer una infraestructura para la interconexión y compartición de recursos de muy diversa naturaleza. El problema consiste en hacer este reparto justo para lo que se debe aplicar un modelo de créditos. El proyecto consiste en el estudio teórico del problema así como en las posibles soluciones, bien sean clásicas o migradas de otros escenarios donde se producen conflictos similares. Además se deberán probar las prestaciones de los modelos propuestos con técnicas de simulación.	Estudiantes de: <ul style="list-style-type: none"> • Ingeniería de Telecomunicaciones • Ingeniería en Informática • Ing. Técnicas Informáticas Gest. • I.T.T. exp. Telemática • I.T.T. exp. Señal e Imagen 	GUERRERO LOPEZ, BLANCA CARMEN (Tutor) CORTES RAMOS, RUBEN (Co-director)	DESPACHO 4 1A04	30-nov-2008	LIBRE
"JAVA'S PURGATORY": DESARROLLO DE UN JUEGO EDUCATIVO EN UNITY3D Creación de un videojuego educativo basado en arquitectura JAVA para enseñar primeros pasos de programación en JAVA, usando el motor UNITY3D.	Estudiantes de: <ul style="list-style-type: none"> • I.T.T. exp. Sistemas de Teles. 	GARCIA NUEDA, JOSE JUAN (Tutor)	nueda@uc3m.es	24-feb-2012	GOMEZ DURAN, DAVID I.T.T. exp. Sistemas de Teles. Asignado: 24-feb-2012
"REALIZACIÓN DE UN "TRAILBLAZER" (TRAZADOR DE PISTAS) INTELIGENTE Y DINÁMICO PARA LA NAVEGACIÓN EN LA WEB" Se trata de diseñar e implementar una aplicación software que, básicamente, facilite la navegación por la Web ofreciendo al usuario posibles enlaces de interés mientras este navega. Para ello tendrá en cuenta, por ejemplo, las páginas visitadas en sesiones previas, las páginas visitadas en la sesión actual, el perfil conocido de usuario, etc.	Estudiantes de: <ul style="list-style-type: none"> • Ingeniería Industrial • Ingeniería de Telecomunicaciones • Ingeniería en Informática • Ing. Técnicas Informáticas Gest. • I.T.T. exp. Sistemas de Teles. • I.T.T. exp. Telemática • I.T.T. exp. Señal e Imagen • Ing. Informática/Computación 	GARCIA NUEDA, JOSE JUAN (Tutor)	nueda@uc3m.es	09-nov-2009	MARTOS TABERNILLA, CRISTINA I.T.T. exp. Telemática Asignado: 23-nov-2009
3D HACIA LA TV INMERSIVA Estudio de sistemas inmersivos mediante el uso de TV en 3D.	Estudiantes de: <ul style="list-style-type: none"> • Ingeniería de Telecomunicaciones • Ingeniería en Informática • I.T.T. exp. Sistemas de Teles. • I.T.T. exp. Señal e Imagen 	ALFARO OLIVERA, DAVID (Tutor)	4-1 A05	23-nov-2013	CHUROS MORENO, ANA I.T.T. exp. Señal e Imagen Asignado: 23-nov-2013
ACCESO WEB CONTROLADO AL EMULADOR DEL MAEMO	Estudiantes de:	BARRIL VILL, PABLO (Tutor)	Por correo a	22-nov-2012	ZURITA DIAZ, CESAR

Figura 9 – Listado de proyectos de un departamento

Aplicación para administración

Se accede a través de la URL <https://aplicaciones.uc3m.es/pfc/LoginAction.do> y sólo se está disponible para profesores y personal de administración de la UC3M. A través de

esta aplicación es posible publicar y gestionar ofertas de PFCs y TFGs que aparecerán reflejadas en la aplicación para alumnos.

De acuerdo con la descripción de la aplicación, ésta sirve:

- Para que los alumnos conozcan toda la oferta de PFC que tienen disponibles.
- Para que los departamentos puedan publicitar sus PFC y llevar un registro de los ofertados, en realización y leídos.
- Para que la administración gestione los PFC.

Hay pues cuatro tipos de usuarios con perfiles de acceso diferentes:

- Profesores
- Directores de departamento
- Secretarías de departamento
- Administración de Alumnos

Aclaraciones sobre terminología:

- **Tutor:** Responsable administrativo del proyecto. Debe haber uno y sólo uno, y debe ser necesariamente un profesor de la uc3m.
- **Director(es):** Persona(s) que ha(n) dirigido el trabajo al alumno. Generalmente habrá un sólo Director que coincidirá con el Tutor, pero pueden ser varios y pueden no ser profesores de la Universidad.
- **Tipos de TFG/PFC:**
 - Internos: Proyectos realizados en la Universidad
 - Externos: Proyectos realizados en empresas, como alumno Erasmus o en otras Universidades (en estos casos, habrá al menos un director externo a la UC3M y que por tanto no sea tutor).

2.4. Sistemas operativos para dispositivos móviles

En esta sección se enumeran los principales sistemas operativos disponibles actualmente para dispositivos móviles:

2.4.1. *Blackberry OS*

Desarrollado por *Research In Motion* para BlackBerry (5). Su sistema permite la realización de multitareas, servicio de mensajería instantánea a través del sistema de PIN a PIN (6). De acuerdo a los fabricantes este sistema operativo móvil está orientado para el desarrollo de tareas profesionales que permiten sincronizar el dispositivo con agenda, correos electrónicos, calendario y contactos.

2.4.2. *iOS*

iOS es un sistema operativo móvil de la empresa *Apple Inc* (7). Originalmente desarrollado para el iPhone (iPhone OS), siendo después usado en dispositivos como el iPod Touch, iPad y el Apple TV. Apple, Inc. no permite la instalación de iOS en hardware de terceros.

iOS se deriva de Mac OS X, que a su vez está basado en Darwin BSD (8), y por lo tanto es un sistema operativo Unix.

2.4.3. *Windows Phone*

Windows Phone es un sistema operativo móvil desarrollado por Microsoft, como sucesor de la plataforma Windows Mobile. A diferencia de su predecesor, está enfocado en el mercado de consumo generalista en lugar del mercado empresarial por lo que carece de muchas funcionalidades que proporcionaba la versión anterior.

El 29 de octubre de 2012 se lanzó al mercado Windows Phone 8 (9) solo para nuevos dispositivos, debido a un cambio completo en el kernel que lo hace incompatible con dispositivos basados en la versión anterior.

2.4.4. *Android*

Sistema Operativo móvil desarrollado por Google para smartphones, tabletas, portátiles, netbooks, Google TV, relojes de pulseras, auriculares y demás dispositivos. Actualmente, existen más de 700 mil aplicaciones (10) de este sistema operativo. En el siguiente apartado se detalla este sistema.

2.5. Sistema operativo Android

Android es un sistema operativo además de una plataforma de software basada en el *kernel* de Linux. Diseñada en un principio para dispositivos móviles, Android permite controlar dispositivos por medio de bibliotecas desarrolladas o adaptadas por Google mediante el lenguaje de programación Java. Por lo tanto se puede decir que Android es una plataforma de código abierto. Esto quiere decir, que cualquier desarrollador puede crear y desarrollar aplicaciones escritas con lenguaje C u otros lenguajes y compilarlas a código nativo de ARM.

2.5.1. *Selección de la plataforma*

La selección de Android como sistema operativo para el desarrollo de la aplicación de gestión de proyectos se ha basado principalmente en los siguientes criterios:

- Cuota de mercado
- Lenguaje de programación

- Tipo de licencia
- Coste de desarrollo/distribución
- Soporte para desarrollo
- Disponibilidad de dispositivos

La Tabla 1 - Comparativa de características en los principales SO para móviles recoge una comparativa de estos aspectos para los sistemas operativos analizados:

S.O.	Cuota (11)	Lenguaje	Licencia	Coste	Soporte	Dispositivos
Blackberry SO	2%	Java	Propietaria	Gratuito	Sí (12)	Limitados a serie Blackberry
iOS	19%	C/C++, Objective-C	Propietaria APSL y Apple EULA	\$99 Anual (Gratis para estudiantes)	Sí (13)	300 millones activos, Limitados a series iPod/iPhone/iPad
Windows Phone	2%	C#	Propietaria Microsoft (EULA)	\$99 Anual (Gratis para estudiantes)	Sí (14)	2 millones, varios fabricantes
Android OS	75%	Java, C/C++ and others (via SL4A)	Apache 2.0 y GNU GPL	\$25 Única cuota	Sí (15)	750 millones activos, varios fabricantes (16)

Tabla 1 - Comparativa de características en los principales SO para móviles

La Figura 10 - Evolución de las ventas de *smartphones* por sistema operativo ilustra la distribución actual de ventas de dispositivos según su sistema operativo.

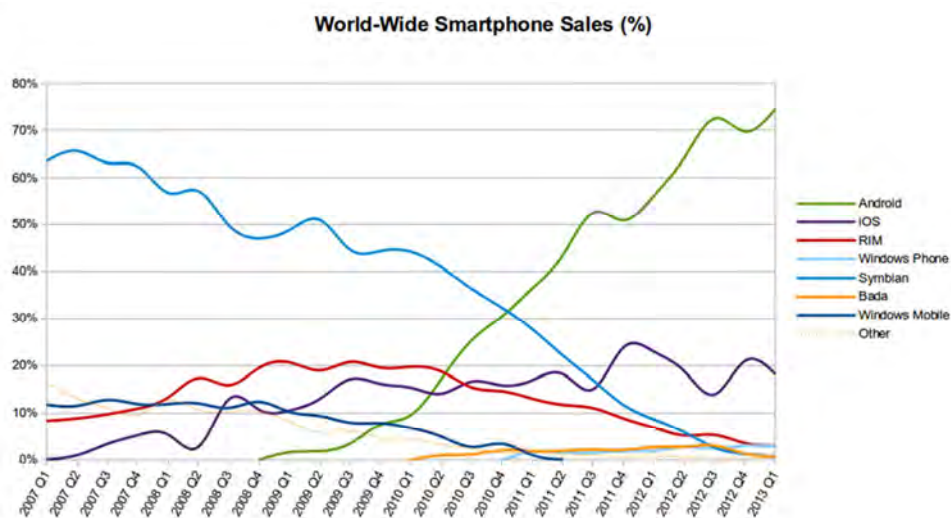


Figura 10 - Evolución de las ventas de *smartphones* por sistema operativo (11)

En vista de estos datos, se ha optado por utilizar Android OS como plataforma para la aplicación:

- Dispone de la mayor cuota de mercado actualmente
- Soporta lenguajes de programación conocidos, como Java.
- Está orientada al código abierto.
- El coste de distribución de aplicaciones es comparativamente menor.
- Dispone de soporte para desarrolladores además de una comunidad creciente de desarrolladores.
- Existen muchos dispositivos disponibles a precios reducidos.

2.5.2. Aspectos generales

Historia y evolución

Fue desarrollado inicialmente por *Android Inc.*, una firma comprada por Google en 2005. El anuncio del sistema Android se realizó el 5 de noviembre de 2007 junto con la creación de la *Open Handset Alliance*, un consorcio de 78 compañías de hardware, software y telecomunicaciones dedicadas al desarrollo de estándares abiertos para dispositivos móviles. Google liberó la mayoría del código de Android bajo la licencia Apache, una licencia libre y de código abierto.

Hubo que esperar hasta 2008 hasta que se produjo el lanzamiento oficial del primer terminal con el sistema operativo móvil de Google. Ese primer *smartphone* Android fue el HTC Dream, que también se comercializó como T-Mobile G1. Con ese dispositivo, que contaba con un teclado Qwerty que se ocultaba tras la pantalla táctil, se iniciaba un catálogo de dispositivos que no ha dejado de crecer, integrando cientos de modelos con una amplia variedad de características.

Versiones

Android ha visto numerosas actualizaciones desde su liberación inicial. Estas actualizaciones al sistema operativo base típicamente arreglan bugs y agregan nuevas funciones. Generalmente cada actualización del sistema operativo Android es desarrollada bajo un nombre en código de un elemento relacionado con postres. Los nombres en código están en orden alfabético.

A continuación se enumeran las versiones de Android liberadas hasta la fecha. Sólo se ha incluido una breve descripción en la más reciente:

- A: Apple Pie (v1.0), Tarta de manzana
- B: Banana Bread (v1.1), Pan de plátano
- C: Cupcake (v1.5), Magdalena glaseada.

- D: Donut (v1.6), Rosquilla.
- E: Éclair (v2.0/v2.1), pastel francés conocido en España como pepito.
- F: Froyo (v2.2), (Abreviatura de _frozen yogurt_) Yogur helado.
- G: Gingerbread (v2.3), Pan de jengibre. El 6 de diciembre del 2010, el SDK de Android 2.3 fue lanzado, basado en el núcleo de linux 2.6.
- H: Honeycomb (v3.0/v3.1/v3.2), Panal de miel. Orientado especialmente a tabletas.
- I: Ice Cream Sandwich (v4.0), Sándwich de helado. Versión que uni_ca el uso en cualquier dispositivo, tanto en teléfonos, tablets, televisores, netbooks, etc.
- J: Jelly Bean (v4.1/v4.1.2/v4.2/v4.2.1), Judía de gominola. Fue lanzada el 9 de julio del 2012, Basada en el núcleo de linux 3.0.31. Nexus 7, el primer dispositivo en correr Jelly Bean, fue lanzado el 13 de julio del 2012. Entre las novedades introducidas destacan:
 - Mayor fluidez con *Project Butter*
 - Notificaciones expandibles
 - Soporte multiusuario para tablets
 - Nuevas funcionalidades como Photo Sphere, Gesture Typing o Google Now.

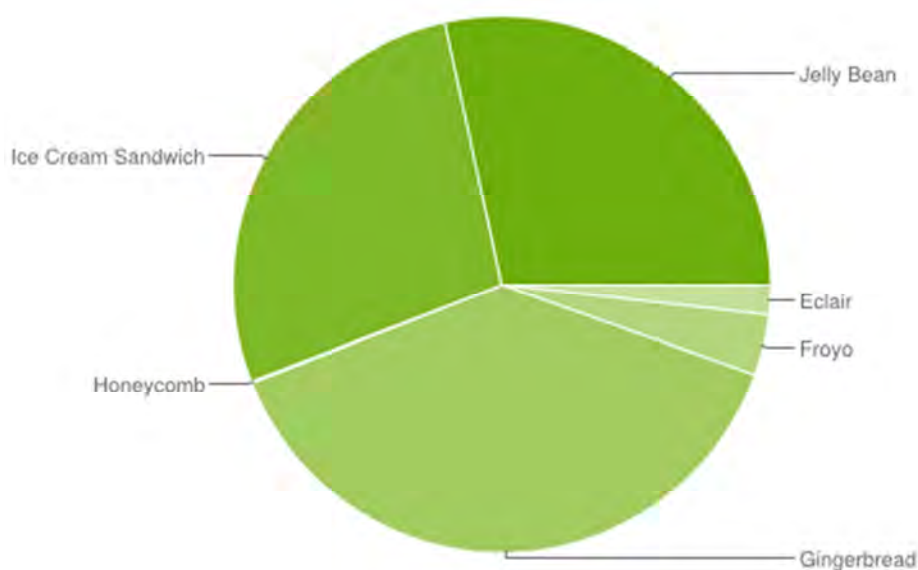


Figura 11 – Distribución del porcentaje de dispositivos Android por versión del OS (17)

Licencia

Una de las grandes cualidades o características de este sistema operativo es su carácter abierto. Android se distribuye bajo dos tipos de licencias, una que abarca todo el código del *kernel* y que es GNU GPLv2 (18) y otra para el resto de componentes del sistema que se licencia bajo APACHE v2 (19).

Estas dos licencias son muy diferentes entre sí y las dos dan forma al sistema para que funcione tal como lo hace respecto a todos los implicados en el ecosistema.

La licencia GNU GPLv2 del *kernel* de Android implica que su código debe ser puesto al alcance de todos y que todos podremos hacer con este código lo que nos parezca oportuno, modificarlo, ampliarlo, recortarlo, pero siempre estaremos en la obligación de volver a licenciarlo con la misma licencia. Google también por supuesto.

En cuanto a la segunda licencia, APACHE v2, y con la cual está licenciado el resto del código del sistema, implica que este código se pueda distribuir para ser modificado y usado a antojo del que lo utilice, pero a diferencia del primer caso, las modificaciones y el código resultante no es obligatorio el licenciarlo bajo las mismas condiciones en las que se encontraba.

Esto en la práctica habilita a los desarrolladores, fabricantes y operadoras a tomar partes del sistema o crear partes nuevas con código creado por ellos pero sin la necesidad de tener que licenciar libremente estos desarrollos. No olvidemos que aunque Android sea gratuito, con él se hace negocio y los fabricantes, operadores y el propio Google invierten mucho dinero en crear dispositivos, sus drivers y aplicaciones o interfaces propias para diferenciar sus dispositivos del resto.

Google libera el código de cada versión del sistema en su repositorio oficial una vez concluye completamente el desarrollo de la versión, esto difiere de algunos otros sistemas que se distribuyen bajo licencias similares y que van liberando partes o secciones del desarrollo global sin tener que esperarse a su conclusión

2.5.3. *Arquitectura*

En las siguientes líneas se dará una visión global por capas de cuál es la arquitectura empleada en Android. Cada una de estas capas utiliza servicios ofrecidos por las anteriores, y ofrece a su vez los suyos propios a las capas de niveles superiores, tal como muestra la Figura 12:



Figura 12 - Arquitectura del Sistema Operativo Android (20)

Aplicaciones

Este nivel contiene, tanto las incluidas por defecto de Android como aquellas que el usuario vaya añadiendo posteriormente, ya sean de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las API y librerías de los niveles anteriores.

Aunque la relación de aplicaciones incluidas por defecto en Android varía entre los distintos fabricantes y dispositivos, comúnmente se pueden encontrar las siguientes:

- Navegador
- Calendario
- Tienda
- Mapas

Desde una aplicación es posible invocar a otra expresamente (por su identificador o nombre de paquete) o simplemente indicar la acción deseada (por ejemplo, enviar un mensaje) de forma que se permita al usuario elegir de entre todas las aplicaciones

instaladas que soportan esa acción.

En el capítulo de Implementación se describe el acceso a las aplicaciones preinstaladas desde la aplicación de gestión de proyectos.

Framework de Aplicaciones

Representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación. Toda aplicación que se desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por Google o terceras compañías, o incluso las que el propio usuario cree, utilizan el mismo conjunto de API y el mismo "framework", representado por este nivel.

Entre las API más importantes ubicadas aquí, se pueden encontrar las siguientes:

- **Activity Manager:** Conjunto de API que gestiona el ciclo de vida de las aplicaciones en Android.
- **Window Manager:** Gestiona las ventanas de las aplicaciones y utiliza la librería Surface Manager.
- **Telephone Manager:** Incluye todas las API vinculadas a las funcionalidades propias del teléfono (llamadas, mensajes, etc.).
- **Content Provider:** Permite a cualquier aplicación compartir sus datos con las demás aplicaciones de Android. Por ejemplo, gracias a esta API la información de contactos, agenda, mensajes, etc. será accesible para otras aplicaciones.
- **View System:** Proporciona un gran número de elementos para poder construir interfaces de usuario (GUI), como listas, mosaicos, botones, "checkboxes", tamaño de ventanas, control de las interfaces mediante teclado, etc. Incluye también algunas vistas estándar para las funcionalidades más frecuentes.
- **Location Manager:** Posibilita a las aplicaciones la obtención de información de localización y posicionamiento.
- **Notification Manager:** Mediante el cual las aplicaciones, usando un mismo formato, comunican al usuario eventos que ocurran durante su ejecución: una llamada entrante, un mensaje recibido, conexión Wi-Fi disponible, ubicación en un punto determinado, etc. Si llevan asociada alguna acción, en Android denominada Intent, (por ejemplo, atender una llamada recibida) ésta se activa mediante un simple clic.
- **XMPP Service:** Colección de API para utilizar este protocolo de intercambio de mensajes basado en XML.

Librerías

La siguiente capa se corresponde con las librerías utilizadas por Android. Éstas han sido escritas utilizando C/C++ y proporcionan a Android la mayor parte de sus capacidades más características. Junto al núcleo basado en Linux, estas librerías constituyen el corazón de Android.

Entre las librerías más importantes ubicadas aquí, se pueden encontrar las siguientes:

- `libc`: Incluye todas las cabeceras y funciones según el estándar del lenguaje C. Todas las demás librerías se definen en este lenguaje.
- `Surface Manager`: Es la encargada de componer los diferentes elementos de navegación de pantalla. Gestiona también las ventanas pertenecientes a las distintas aplicaciones activas en cada momento.
- `OpenGL/S`L y `SGL`: Representan las librerías gráficas y, por tanto, sustentan la capacidad gráfica de Android. `OpenGL/S`L maneja gráficos en 3D y permite utilizar, en caso de que esté disponible en el propio dispositivo móvil, el hardware encargado de proporcionar gráficos 3D. Por otro lado, `SGL` proporciona gráficos en 2D, por lo que será la librería más habitualmente utilizada por la mayoría de las aplicaciones. Una característica importante de la capacidad gráfica de Android es que es posible desarrollar aplicaciones que combinen gráficos en 3D y 2D.
- `Media Libraries`: Proporciona todos los códecs necesarios para el contenido multimedia soportado en Android (vídeo, audio, imágenes estáticas y animadas, etc.)
- `FreeType`: Permite trabajar de forma rápida y sencilla con distintos tipos de fuentes.
- `SSL`: Posibilita la utilización de dicho protocolo para establecer comunicaciones seguras.
- `SQLite`: Creación y gestión de bases de datos relacionales.
- `WebKit`: Proporciona un motor para las aplicaciones de tipo navegador y forma el núcleo del actual navegador incluido por defecto en la plataforma Android.

2.5.4. *Desarrollo con Android*

Entorno de desarrollo

En este apartado se describen de forma muy resumida los componentes de un entorno de desarrollo de aplicaciones Android:

- *Java Development Kit (JDK)*, descargable desde la web de Oracle para distintos SO y arquitecturas de procesador.

- *Android Standard Development Kit (Android SDK)*. Se puede obtener desde la página de desarrolladores de Android.
- Entorno de Desarrollo Integrado (IDE). Eclipse es uno de los entornos de desarrollo integrado (IDE, por sus siglas en inglés) más populares. Puede utilizarse para desarrollar aplicaciones en distintos lenguajes mediante extensiones o "plugins", incluido Android. Puede obtenerse desde la web de Eclipse.
- *Android Development Tools (ADT)*. Google pone a disposición de los desarrolladores un plugin para Eclipse llamado *Android Development Tools (ADT)* que facilita en gran medida el desarrollo de aplicaciones para la plataforma. Puede descargarse directamente desde el diálogo de instalación de software de Eclipse.
- *Android Virtual Device (AVD)*: ADT incluye un emulador o dispositivo virtual donde el desarrollador puede probar y depurar sus aplicaciones.

Distribución

Las aplicaciones para Android se distribuyen principalmente a través de Google Play (antes llamado *Android Market*) es una tienda de software en línea desarrollada por Google para los dispositivos Android. Es una aplicación ("app") que está preinstalada en la mayoría de los dispositivos Android y que permite a los usuarios buscar y descargar aplicaciones publicadas por desarrolladores terceros, alojada en Google Play. Los usuarios también pueden buscar y obtener información sobre aplicaciones en esta página web. A finales de 2013, Google Play disponía de alrededor de un millón de aplicaciones (21).

2.6. Servicios Web

Un servicio web (en inglés, *Web services*) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones (22). Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios Web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares. Es una máquina que atiende las peticiones de los clientes web y les envía los recursos solicitados.

2.6.1. Protocolos

En esta sección compararemos dos de los principales protocolos para servicios web: SOAP y REST.

SOAP

SOAP (siglas de *Simple Object Access Protocol*) es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML (23). Este protocolo deriva de un protocolo creado por David Winer en 1998, llamado XML-RPC. SOAP fue creado por Microsoft, IBM y otros y está actualmente bajo el auspicio de la W3C. Es uno de los protocolos utilizados en los servicios Web.

SOAP puede formar la capa base de una "pila de protocolo de *web service*", ofreciendo un *framework* de mensajería básica en la cual los *web services* se puedan construir. Este protocolo basado en XML consiste de tres partes: un sobre (*envelope*), el cual define qué hay en el mensaje y cómo procesarlo; un conjunto de reglas de codificación para expresar instancias de tipos de datos; y una conversión para representar llamadas a procedimientos y respuestas. El protocolo SOAP tiene tres características principales:

- Extensibilidad (seguridad y WS-routing son extensiones aplicadas en el desarrollo).
- Neutralidad (SOAP puede ser utilizado sobre cualquier protocolo de transporte como HTTP, SMTP, TCP o JMS).
- Independencia (SOAP permite cualquier modelo de programación).

REST

REST (siglas de *REpresentational State Transfer*) define un set de principios arquitectónicos por los cuales se diseñan servicios web haciendo foco en los recursos del sistema, incluyendo cómo se accede al estado de dichos recursos y cómo se transfieren por HTTP hacia clientes escritos en diversos lenguajes. REST emergió en los últimos años como el modelo predominante para el diseño de servicios. De hecho, REST logró un impacto tan grande en la web que prácticamente logró desplazar a SOAP y las interfaces basadas en WSDL por tener un estilo bastante más simple de usar.

REST no tuvo mucha atención cuando Roy Fielding lo presentó por primera vez en el año 2000 en la Universidad de California, durante la charla académica *Estilos de Arquitectura y el Diseño de Arquitecturas de Software basadas en Redes* (24), la cual analizaba un conjunto de principios arquitectónicos de software para usar a la Web como una plataforma de Procesamiento Distribuido. Ahora, años después de su presentación, comienzan a aparecer varios *frameworks* REST y se convirtió en una parte integral de Java 6 a través de JSR-311 (25).

Una implementación concreta de un servicio web REST sigue cuatro principios de diseño fundamentales:

- Utiliza los métodos HTTP de manera explícita
- No mantiene estado
- Expone URIs “amigables” (con forma de directorios)
- Transfiere XML, JavaScript Object Notation (JSON), o ambos

2.6.2. Formatos

XML

XML, siglas en inglés de *eXtensible Markup Language* ('lenguaje de marcas extensible'), es un lenguaje de marcas desarrollado por el W3C (26). Deriva del lenguaje SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML) para estructurar documentos grandes. A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones se deben comunicar entre sí o integrar información. XML el formato estándar para el intercambio de datos en servicios web.

JSON

JSON, acrónimo de *JavaScript Object Notation*, es un formato ligero para el intercambio de datos. JSON es un subconjunto de la notación literal de objetos de JavaScript que no requiere el uso de XML (27). La simplicidad de JSON ha dado lugar a la generalización de su uso, especialmente como alternativa a XML en AJAX (28). Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador sintáctico (parser) de JSON.

<p>http://localhost:8080/Json/SyncReply/Contacts</p> <pre> { - Contacts: [- { FirstName: "Demis", LastName: "Bellot", Email: "demis.bellot@gmail.com" }, - { FirstName: "Steve", LastName: "Jobs", Email: "steve@apple.com" }, - { FirstName: "Steve", LastName: "Ballmer", Email: "steve@microsoft.com" }, - { FirstName: "Eric", LastName: "Schmidt", Email: "eric@google.com" }, - { FirstName: "Larry", LastName: "Ellison", Email: "larry@oracle.com" }] } </pre>	<p>http://localhost:8080/Xml/SyncReply/Contacts</p> <pre> <ContactsResponse xmlns:i="http://www.w3.org/20 <Contacts> <Contact> <Email>demis.bellot@gmail.com</Email> <FirstName>Demis</FirstName> <LastName>Bellot</LastName> </Contact> <Contact> <Email>steve@apple.com</Email> <FirstName>Steve</FirstName> <LastName>Jobs</LastName> </Contact> <Contact> <Email>steve@microsoft.com</Email> <FirstName>Steve</FirstName> <LastName>Ballmer</LastName> </Contact> <Contact> <Email>eric@google.com</Email> <FirstName>Eric</FirstName> <LastName>Schmidt</LastName> </Contact> <Contact> <Email>larry@oracle.com</Email> <FirstName>Larry</FirstName> <LastName>Ellison</LastName> </Contact> </Contacts> </ContactsResponse> </pre>
--	---

Figura 13 – Comparación de una respuesta JSON (izquierda) y XML (derecha) con la misma información (29)

En JavaScript, un texto JSON se puede analizar fácilmente usando el procedimiento `eval()`, lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la

comunidad de desarrolladores AJAX, debido a la ubicuidad de JavaScript en casi cualquier navegador web (30). La figura 13 muestra un ejemplo entre JSON y XML.

2.7. Marco regulatorio

En este apartado se revisa la normativa legal vigente que regula la seguridad en el tratamiento de datos personales para su consideración en el diseño e implementación de la aplicación.

2.7.1. Normativa general sobre protección de datos personales

La Ley Orgánica 15/1999 de 13 de diciembre de Protección de Datos de Carácter Personal, (LOPD), es una Ley Orgánica española que tiene por objeto garantizar y proteger, en lo que concierne al tratamiento de los datos personales, las libertades públicas y los derechos fundamentales de las personas físicas, y especialmente de su honor, intimidad y privacidad personal y familiar (31).

Su objetivo principal es regular el tratamiento de los datos y ficheros, de carácter personal, independientemente del soporte en el cual sean tratados, los derechos de los ciudadanos sobre ellos y las obligaciones de aquellos que los crean o tratan. Los dos aspectos más destacables de esta ley son el derecho de información en la recogida de datos (Artículo 5) y el derecho de consentimiento por parte del afectado (Artículo 6).

2.7.2. Normativa sobre protección de datos personales en aplicaciones móviles

Ante la generalización del empleo de aplicaciones en dispositivos móviles, el llamado Grupo de trabajo europeo sobre protección de datos del artículo 29 (órgano asesor independiente formado básicamente por las autoridades nacionales en la materia) dependiente de la Comisión Europea ha publicado su Dictamen 2/2013, titulado en su versión en inglés (la única de momento disponible) “Opinion 02/2013 on apps on smart devices” (32).

El Dictamen finaliza con unas conclusiones muy detalladas y recomendaciones dirigidas tanto a los fabricantes de las aplicaciones, comúnmente conocidos como desarrolladores, a los vendedores o distribuidores de las mismas y a los fabricantes de los sistemas operativos y los dispositivos en los que se descargan esas aplicaciones. El Grupo advierte que existe un solapamiento de responsabilidades en la protección de datos entre las diferentes partes que participan en el desarrollo, la distribución y la capacidad técnica de las aplicaciones.

La mayoría de las conclusiones y recomendaciones se dirigen a los desarrolladores de aplicaciones (ellos son los que mayor control tienen sobre la forma precisa en que se realiza el tratamiento o en que se presenta la información dentro de la aplicación), pero, para alcanzar el máximo nivel de protección de la intimidad y protección de datos, deben colaborar con otras partes del ecosistema de las aplicaciones, como los fabricantes de

sistemas operativos y dispositivos, las tiendas de aplicaciones y terceras partes como los proveedores de análisis y las redes publicitarias (33).

2.7.3. Seguridad de datos en aplicaciones Android

Android proporciona a los desarrolladores una serie de recomendaciones en materia de seguridad (34). A continuación se enumeran algunos de los aspectos más destacables.

Almacenamiento de datos

Por defecto toda la información generada por la aplicación se almacena de forma interna a ésta por lo que no puede ser accedida por otras aplicaciones. Android considera que este nivel de protección es en general suficiente para la mayoría de aplicaciones.

Permisos

Debido al aislamiento que existe entre las aplicaciones Android, éstas deben declarar explícitamente los permisos que requieren para el uso de capacidades adicionales tales como el acceso a red, a ficheros o a la cámara.

Gestión de datos de usuario y credenciales

La principal recomendación de Android en este sentido es minimizar el manejo de datos personales y evitar su almacenamiento en la medida de lo posible. Asimismo delega en la normativa vigente en cada país la obligatoriedad de informar al usuario sobre el uso de estos datos.

3. Análisis del sistema

3.1. Introducción

Este capítulo se corresponde con la Especificación de Requisitos Software (ERS) para la aplicación de administración de proyectos de fin de carrera de la Universidad Carlos III de Madrid. La especificación ha sido elaborada tomando en cuenta la funcionalidad disponible actualmente en la versión Web online, así como las mejoras posibles aprovechando las posibilidades que ofrecen aplicaciones móviles.

3.1.1. *Propósito*

El objetivo de esta especificación es definir de manera clara y precisa las funcionalidades y restricciones que tendrá el sistema que se desea construir, y va dirigida tanto a los responsables de evaluar el proyecto como a las personas que harán uso del sistema terminado.

3.1.2. *Ámbito del sistema*

El sistema que se desea construir permitirá la gestión de todas las tareas relacionadas con el trabajo de fin de grado, desde su búsqueda, selección y matrícula hasta su desarrollo, seguimiento y presentación.

La aplicación está en principio orientada a su uso por estudiantes, si bien se contempla su extensión para añadir funcionalidades específicas para tutores.

Este nuevo sistema no pretende la sustitución de la aplicación web actual, sino la integración con la misma para ofrecer funcionalidades comunes y complementarias aprovechando las posibilidades que ofrecen las aplicaciones móviles.

3.2. Descripción general

3.2.1. *Capacidades generales*

En este apartado se comenta cuál es el objetivo que debe de cumplir el sistema de cara a los usuarios. Por ello a continuación se muestran cuales son las capacidades que se han de cumplir de cara al análisis realizado inicialmente, que en sucesivos apartados se detallará mas profundamente.

- **Información general:** Datos de interés sobre los PFC incluyendo entre otros normativa, instrucciones de matriculación y plazos.
- **Tablón:** Listado de los proyectos disponibles clasificados por departamentos. Para cada proyecto disponible, se presentarán sus datos más relevantes como la descripción, disponibilidad y tutor.

- **Búsqueda:** Permite a los usuarios consultar los proyectos filtrando por texto y otros criterios tales como departamento, tutor y fecha.
- **Agenda:** Almacena los eventos creados por el usuario en relación con el PFC (reuniones, entregas, etc.). El usuario recibirá avisos cuando se aproxime la fecha del evento. Se contempla que la agenda pueda ser compartida entre el estudiante y el tutor.
- **Perfil:** Permite al estudiante añadir y editar sus datos personales y preferencias sobre proyecto para su uso por otras funcionalidades de la aplicación.
- **Matrícula:** Permite al estudiante matricular el proyecto seleccionado a través de los mecanismos ofrecidos por el servidor de la UC3M.
- **Contacto:** Mecanismo de comunicación directa entre el estudiante y el tutor. El contenido de las comunicaciones será almacenado para su consulta por el usuario.
- **Seguimiento:** Representa de forma sencilla e inmediata el estado de avance del PFC en sus distintas fases.
- **Documentos:** Sistema de edición y compartición de documentos entre el estudiante y el tutor.
- **Ajustes:** El usuario podrá personalizar ciertos aspectos de la aplicación así como almacenar sus PFC favoritos durante la búsqueda.

3.2.2. Restricciones generales

En este apartado se van a comentar las restricciones que se van a dar inicialmente en el sistema, que son necesarias para diseñar el sistema de control de la gestión de la biblioteca.

- **Seguridad:** Las funciones que requieran acceso a Aula Global estarán sometidas a los mismos criterios de acceso y seguridad que la web de la UC3M.
- **Dependencias:** La aplicación se desarrollará para un SO y una versión específica del mismo. No obstante, su planificación e implementación debe facilitar en la medida de lo posible su migración a otros SO. Del mismo modo, la integración con los servicios de la UC3M se realizará de forma que admitan modificaciones en los mismos mediante cambios sencillos de la aplicación.
- **Accesibilidad:** Se proporcionará traducción al menos a los idiomas soportados actualmente por el web de la UC3M. Adicionalmente, se facilitará en la medida de lo posible el acceso a personas con facultades visuales reducidas mediante el ajuste de tamaño de texto.
- **Modularidad:** La aplicación debe diseñarse e implementarse de forma modular, en previsión de que pueda ser extendida o integrada en otras

aplicaciones en un futuro.

3.2.3. Características de los usuarios

Cualquier persona que disponga de un terminal móvil compatible (Ver ANEXO B.1 Requisitos mínimos), con conexión a internet y acceso al sitio de descarga podrá instalar y utilizar la aplicación.

No obstante, la aplicación está orientada a estudiantes y tutores de la UC3M a los que se les presupone un conocimiento de la estructura y funcionamiento de esta Universidad. Algunas funciones de la aplicación sólo están disponibles si el usuario tiene acceso a Aula Global.

3.3. Requisitos de usuario

El objetivo de este apartado es detallar los requisitos de usuario para poder diseñar en futuros apartados la aplicación de gestión de PFCs y TFGs. Estos requisitos se dividen en de capacidad y de restricción. Los primeros indican qué debe ser capaz de realizar el sistema y los segundos especifican las restricciones que establecen el cómo se realizan las tareas que fueron indicadas en los primeros. El formato utilizado en la especificación se muestra en la Tabla 2 cuyos campos son los siguientes:

- **Identificador:** es el código único que identifica a cada requisito.
- **Título:** es el nombre único del requisito.
- **Descripción:** detalla la especificación del requisito de manera precisa y simple.
- **Prioridad:** establece la importancia del requisito en función del desarrollo del proyecto. Los valores pueden ser alta, media o baja.

Identificador	XX-UR-YY
Título	
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	

Tabla 2 - Plantilla de requisito de usuario

3.3.1. Requisitos de capacidad

Identificador	RUC-01
Título	Buscar proyectos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario es capaz de buscar proyectos a partir de los criterios introducidos.	

Tabla 3: RUC-01

Identificador	RUC-02
Título	Filtrar los resultados obtenidos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
En el listado de proyectos encontrados, el usuario puede filtrar los resultados aplicando criterios adicionales.	

Tabla 4: RUC-02

Identificador	RUC-03
Título	Acceder a los detalles de un proyecto
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario puede acceder a los detalles del proyecto seleccionado en los listados de resultados de búsqueda, búsquedas recientes o favoritas.	

Tabla 5: RUC-03

Identificador	RUC-04
Título	Contactar con el tutor del proyecto
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
Desde el detalle del proyecto, el usuario podrá contactar con el tutor a cargo del mismo para solicitar más información.	

Tabla 6: RUC-04

Identificador	RUC-05
Título	Crear un perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario es capaz de crear un perfil con sus datos personales y sus preferencias para la selección de proyectos. Puede editar sus preferencias en cualquier momento.	

Tabla 7: RUC-05

Identificador	RUC-06
Título	Encontrar los proyectos que mejor se ajustan al perfil
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
Una vez completado su perfil, el usuario puede realizar búsquedas que se ajusten a los datos proporcionados.	

Tabla 8: RUC-06

Identificador	RUC-07
Título	Acceder a la información oficial sobre proyectos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario puede consultar la normativa y otra documentación oficial de la UC3M sobre proyectos desde la propia aplicación y en un formato adaptado al dispositivo.	

Tabla 9: RUC-07

Identificador	RUC-08
Título	Recuperar búsquedas recientes
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
La aplicación permite al usuario consultar y volver a realizar búsquedas de proyectos recientes.	

Tabla 10: RUC-08

Identificador	RUC-09
Título	Acceder a la información del proyecto matriculado
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario puede obtener información del proyecto que haya matriculado (es decir, en el que figura como alumno).	

Tabla 11: RUC-09

Identificador	RUC-10
Título	Administrar eventos del calendario sobre proyectos
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario puede crear eventos en el calendario relacionados con proyectos (citas con tutor, entregas, presentación...)	

Tabla 12: RUC-10

Identificador	RUC-11
Título	Acceder a la ayuda de la aplicación
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario dispone de una sección donde se explica el funcionamiento de la aplicación.	

Tabla 13: RUC-11

Identificador	RUC-12
Título	Compartir documentos con el tutor del proyecto
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario puede compartir documentos relacionados con el proyecto con su tutor.	

Tabla 14: RUC-12

Identificador	RUC-13
Título	Marcar proyectos como favoritos
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
Los proyectos encontrados pueden ser marcados como favoritos por el usuario para ser consultados posteriormente.	

Tabla 15: RUC-13

Identificador	RUC-14
Título	Compartir la información del proyecto
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Descripción	
El usuario puede compartir con sus contactos los detalles de un proyecto.	

Tabla 16: RUC-14

Identificador	RUC-15
Título	Matricular un proyecto
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Descripción	
El usuario puede matricular un proyecto a través de la aplicación.	

Tabla 17: RUC-15

Identificador	RUC-16
Título	Informar al tutor de la evolución del proyecto
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja
Descripción	
El usuario puede compartir documentos relacionados con el proyecto con su tutor.	

Tabla 18: RUC-16

3.3.2. Requisitos de restricción

Identificador	RUR-01
Título	Compatibilidad
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario puede utilizar la aplicación en el mayor número posible de dispositivos.	

Tabla 19: RUR-01

Identificador	RUR-02
Título	Acceso a información remota
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario puede acceder a datos que no se encuentran almacenados en el dispositivo, sino que están disponibles en un servidor remoto.	

Tabla 20: RUR-02

Identificador	RUR-03
Título	Tiempos de espera
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El tiempo que debe esperar el usuario entre que realiza una acción (ej. consulta) y obtiene la respuesta debe ser lo más reducido posible.	

Tabla 21: RUR-03

Identificador	RUR-04
Título	Rendimiento
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
Al utilizar la aplicación en usuario no debe percibir una disminución del rendimiento del dispositivo.	

Tabla 22: RUR-04

Identificador	RUR-05
Título	Protección de datos
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El manejo de los datos personales introducidos por el usuario debe ser conforme a lo recogido el marco regulatorio.	

Tabla 23: RUR-05

Identificador	RUR-06
Título	Extensibilidad
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario podrá ver incrementadas las funcionalidades de la aplicación en un futuro mediante actualizaciones que no afecten a los datos ya almacenados.	

Tabla 24: RUR-06

Identificador	RUR-07
Título	Idioma de la aplicación
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
El usuario podrá utilizar la aplicación al menos en español y en inglés.	

Tabla 25: RUR-07

Identificador	RUR-08
Título	Accesibilidad
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	
Un usuario con problemas de accesibilidad (especialmente visual) podrá utilizar atajos de teclado o incrementar el tamaño de la fuente.	

Tabla 26: RUR-08

3.4. Requisitos de software

En este apartado vamos a detallar los requisitos de software, tanto funcionales como no funcionales, obtenidos a través del análisis exhaustivo de los requisitos de usuario y de la información extraída por el cliente. Los tipos de requisitos que vamos a detallar son los siguientes:

- **Requisitos de software funcional:** Especifican el “qué” tiene que hacer el software, es decir, el propósito del software.
- **Requisitos de software no funcional:** Especifican el “cómo” tiene que hacerse y funcionar el software.

Los requisitos de software se enumeran a continuación como RS(T)-XX. El formato seguido en la especificación de los requisitos es el mismo que el definido previamente en la Tabla 2, añadiendo el campo de Fuente para indicar los requisitos de usuario de los que se deriva cada requisito de software.

3.4.1. Requisitos de software funcional

Identificador	RSF-01		
Título	Realizar búsquedas de departamentos		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-01, RUC-02		
Descripción	El software permite recuperar el listado de departamentos de la UC3M consultando al servidor.		

Tabla 27: RSF-01

Identificador	RSF-02		
Título	Realizar búsquedas de titulaciones		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-01, RUC-02		
Descripción	El software permite recuperar el listado de titulaciones de la UC3M consultando al servidor.		

Tabla 28: RSF-02

Identificador	RSF-03		
Título	Realizar búsquedas de proyectos		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-01, RUC-02, RUC-06		
Descripción	El software puede recuperar el listado de proyectos a partir de los criterios indicados consultando al servidor.		

Tabla 29: RSF-03

Identificador	RSF-04		
Título	Acceder a la información de un proyecto		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-03, RUC-09		
Descripción	El software permite recuperar la información de un proyecto concreto consultando al servidor.		

Tabla 30: RSF-04

Identificador	RSF-05		
Título	Realizar búsquedas de directorio		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-04, RUC-12		
Descripción	El software permite recuperar los datos de un profesor de la UC3M.		

Tabla 31: RSF-05

Identificador	RSF-06		
Título	Acceder a información disponible en la web		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-07		
Descripción	El software permite recuperar la información contenida en una página web y mostrarla en un formato adaptado al dispositivo.		

Tabla 32: RSF-06

Identificador	RSF-07		
Título	Gestionar datos del perfil de usuario		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-05		
Descripción	El software permite gestionar información personalizada para el usuario.		

Tabla 33: RSF-07

Identificador	RSF-08		
Título	Gestionar datos del proyecto del usuario		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-09, RUC-15		
Descripción	El software permite gestionar información personalizada para el proyecto del usuario.		

Tabla 34: RSF-08

Identificador	RSF-09		
Título	Gestionar datos sobre la actividad del usuario		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-08, RUC-13		
Descripción	El software permite registrar datos de la actividad del usuario, tales como consultas recientes.		

Tabla 35: RSF-09

Identificador	RSF-10		
Título	Añadir eventos a la agenda		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-10		
Descripción	El software permite gestionar eventos en el calendario.		

Tabla 36: RSF-10

Identificador	RSF-11		
Título	Envío de información		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-12, RUC-14, RUC-16		
Descripción	El software enviar información (mensajes o documentos).		

Tabla 37: RSF-11

Identificador	RSF-12		
Título	Acceso y navegación por la ayuda		
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-11		
Descripción	El software permite acceder y navegar por la ayuda de la aplicación.		

Tabla 38: RSF-12

Identificador	RSF-13		
Título	Realizar operaciones autenticadas		
Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUC-15		
Descripción	El software realizar operaciones que requieran seguridad en los datos manejados (ej. matriculación)		

Tabla 39: RSF-13

3.4.2. Requisitos de software no funcional

Identificador	RSNF-01		
Título	Compatibilidad con Android 2.3 o superior		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-01		
Descripción	Teniendo en cuenta los datos de la Figura 11 – Distribución del porcentaje de dispositivos Android por versión del OS se considera la versión 2.3 (Gingerbread) como la más extendida. La API utilizada en el desarrollo debe por tanto garantizar la compatibilidad con Android 2.3.		

Tabla 40: RSNF-01

Identificador	RSNF-02		
Título	Realización de consultas remotas al servidor		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-02, RUR-03		
Descripción	El software permite realizar consultas al servidor utilizando autenticación, procesar el contenido y mostrarlo en un formato adaptado al dispositivo. Las consultas deben realizarse de un modo eficiente reduciendo los tiempos de espera.		

Tabla 41: RSNF-02

Identificador	RSNF-03		
Título	Almacenamiento de datos		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03, RUR-04		
Descripción	El software permite la persistencia de datos entre sesiones. Los datos deben almacenarse y recuperarse de forma eficiente.		

Tabla 42: RSNF-03

Identificador	RSNF-04		
Título	Seguridad		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-03, RUR-04		
Descripción	Los datos del usuario no serán compartidos a otras aplicaciones. Su envío a terceros, si fuese necesario, se realizará mediante mecanismos que garanticen su integridad. Se informará al usuario de esta política.		

Tabla 43: RSNF-04

Identificador	RSNF-05		
Título	Modularidad		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-05		
Descripción	La aplicación debe diseñarse de forma modularizada de modo que permita la actualización total o parcial de funcionalidades en el futuro garantizando la integridad de los datos del usuario.		

Tabla 44: RSNF-05

Identificador	RSNF-06		
Título	Soporte multidioma		
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-06		
Descripción	El software debe soportar la traducción a varios idiomas. La adaptación no se limita únicamente a la traducción, sino al formato de fechas, etc.		

Tabla 45: RSNF-06

Identificador	RSNF-07		
Título	Accesibilidad		
Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificable	<input checked="" type="checkbox"/> Sí <input type="checkbox"/> No
Fuente	RUR-07		
Descripción	El software debe soportar opciones para mejorar la accesibilidad de la aplicación.		

Tabla 46: RSNF-07

4. Diseño

En este capítulo se describe el diseño detallado de la aplicación. Además se ofrece una explicación de cada una de las decisiones tomadas a la hora de realizar el diseño.

4.1. Arquitectura del sistema

El diseño de la aplicación se basará en dos patrones de arquitectura muy extendidos: el esquema Cliente-Servidor y el esquema Modelo-Vista-Controlador. Estos patrones se describen más detalladamente a continuación.

4.1.1. Modelo Cliente – Servidor

Es un modelo en el cual las tareas se reparten entre los servidores y los llamados clientes. Un cliente realiza peticiones a otro programa y el servidor es el que se encarga de enviarle la respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre un solo ordenador, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

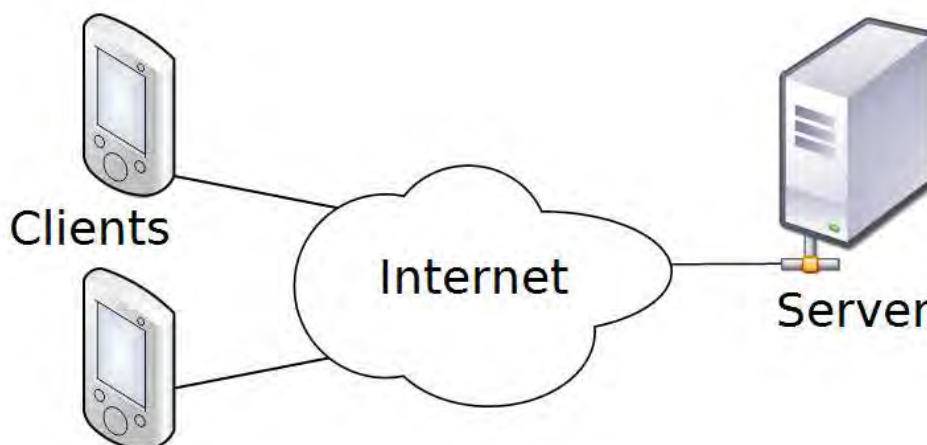


Figura 14 - Arquitectura Cliente-Servidor (35)

- **Cliente:** maneja todas las funciones relacionadas con la manipulación y despliegue de datos, por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario (GUI), además de acceder a los servicios distribuidos en cualquier parte de una red.
- **Servidor:** es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se le conoce con el término *backend*. Normalmente el servidor maneja todas las

funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos.

4.1.2. *Modelo Vista Controlador (MVC)*

El *Modelo Vista Controlador* es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario (GUI), y la lógica de negocio en tres componentes distintos. El patrón MVC, se aprecia frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

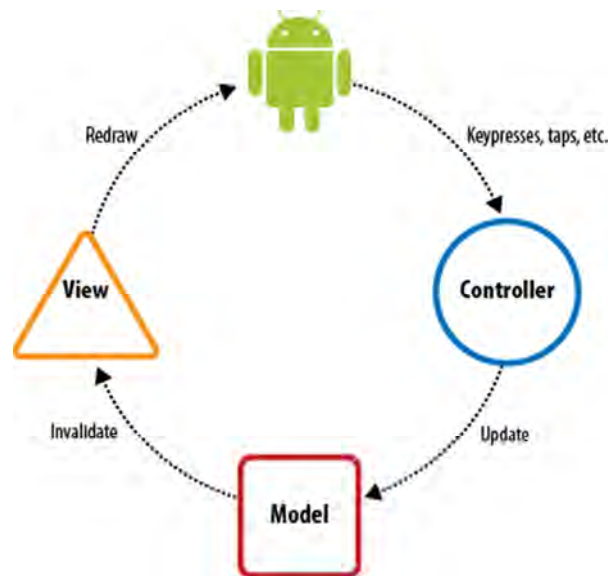


Figura 15 - Arquitectura MVC aplicada al diseño de aplicaciones Android (36)

De una forma más simple, se puede el patrón MVC como un patrón arquitectural que describe una forma de desarrollar aplicaciones software separando los componentes en tres grupos, los cuales explicamos a continuación:

- **Modelo:** es la representación específica de la información con la cual el sistema opera. En resumen, el modelo se limita a lo relativo de la vista y su controlador facilitando las presentaciones visuales complejas. El sistema también puede operar con más datos no relativos a la presentación, haciendo uso integrado de otras lógicas de negocio y de datos afines con el sistema modelado.
- **Vista:** presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario (GUI).
- **Controlador:** responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y, probablemente a la vista.

4.1.3. Uso de los modelos en la aplicación de proyectos

En el caso de la aplicación Android para la gestión de proyectos, ésta actuará como *cliente* mientras que el servicio web de proyectos de la UC3M (Ver Diseño de la aplicación servidora) realizará la función de *servidor*. La comunicación entre ambos puntos se realizará mediante peticiones HTTP con un esquema de datos JSON.

Dentro de la propia aplicación cliente, el *Modelo* corresponde a los datos que deben almacenarse en la aplicación (perfil del usuario, consultas recientes, favoritos, preferencias...). Los datos del servidor también constituyen parte del *Modelo*. Las *Vistas* son las distintas interfaces de usuario que ofrece la aplicación. Y el *Controlador* en este caso es el código que controla el acceso al modelo de datos, tanto al que reside en la aplicación cliente como en el servidor remoto.

La Figura 18 ilustra el empleo de los esquemas analizados en la arquitectura de la aplicación:

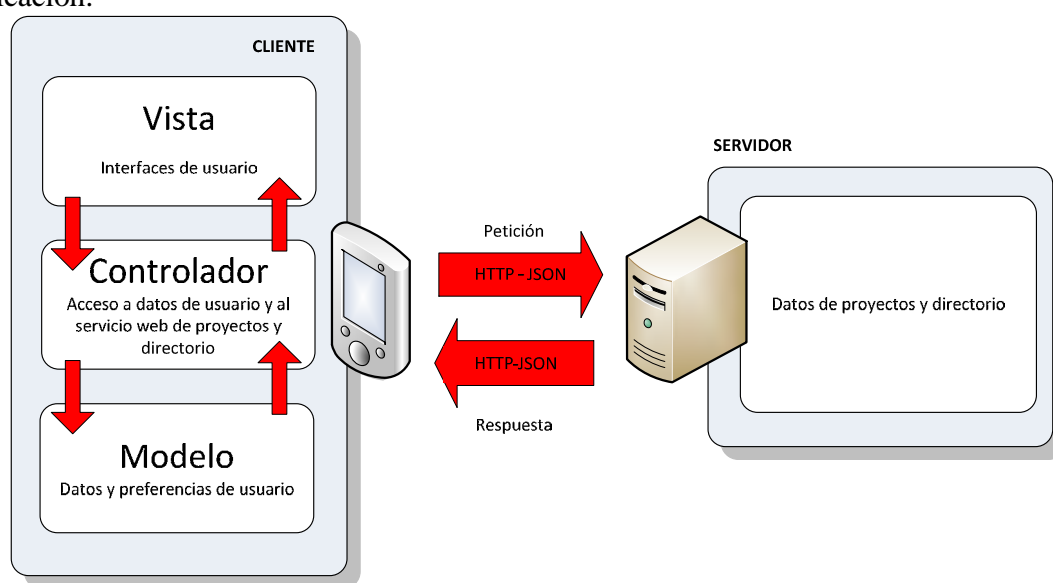


Figura 16 - Arquitectura completa del sistema

Según el diagrama mostrado anteriormente, el flujo de datos en la aplicación se realiza de la siguiente manera:

- El usuario de la aplicación cliente realiza una consulta a través de la interfaz gráfica.
- En el caso de una consulta sobre datos remotos (proyectos, directorio...), el controlador de la aplicación cliente realiza una petición HTTP al servidor, donde su propio controlador extrae la información del modelo de datos del servidor. El servidor crea una respuesta HTTP que incluye la respuesta JSON, enviándolo a la dirección desde la que recibió la petición, que en este caso es el dispositivo móvil.
- En el caso de una consulta sobre datos de la propia aplicación (perfil de usuario, preferencias...) el controlador de la aplicación cliente realizará una consulta al modelo de datos interno de la aplicación.

- En ambos casos, el controlador de la aplicación cliente procesará los datos del modelo para entregarlos a la capa de vista en el formato adecuado.

En los siguientes apartados de este capítulo se analizará en detalle el diseño de la arquitectura de la aplicación.

4.2. Diseño de la aplicación cliente

En este apartado se detalla el diseño de la aplicación cliente. La descripción cubre cada una de las capas que conforman el patrón *Modelo Vista Controlador* estudiado anteriormente.

4.2.1. Interfaces de usuario

En este apartado se describe el diseño de la capa de **Vista**.

Vista de escritorio

El acceso a la aplicación se realiza mediante su icono en el escritorio o en el gestor de aplicaciones.

Autenticación

El usuario debe autenticarse antes de acceder a los contenidos de la aplicación.

Menú principal

Esta vista consiste en un menú de tipo rejilla con iconos de acceso a otras vistas de la aplicación, como se describe en el diagrama de flujo de vistas de la aplicación. Mediante los ajustes de la aplicación será posible añadir o quitar elementos de este menú.

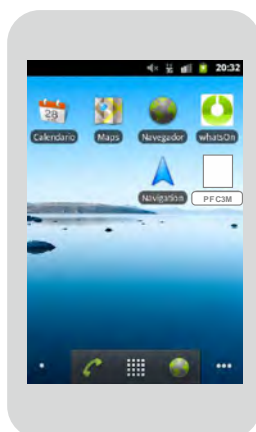


Figura 17 - Vista de escritorio



Figura 18 - Vista de acceso



Figura 19 - Vista de menú principal

Tablón

Muestra un listado con los departamentos que ofrecen PFC. Junto al nombre del departamento se indica el número de proyectos no adjudicados. Al pulsar sobre un departamento, se accede al listado con sus proyectos.

Listado de proyectos

A esta vista se puede llegar desde el tablón o desde el buscador. Consiste en un listado en el que cada elemento representa un PFC. El color del ítem representa el estado del PFC, para una rápida identificación del mismo. Además se incluye su título, su tutor responsable y su fecha de publicación.

A través de la barra de menú inferior es posible realizar un filtrado rápido del resultado mostrando sólo los libres, los marcados como favoritos o aquéllos que más se ajusten al perfil del usuario. Asimismo se puede acceder al buscador para un filtrado más avanzado. Finalmente, pulsando sobre un PFC se accede a su vista de detalle.

Detalle de proyecto

Contiene toda la información relevante de un PFC. Desde la barra de menú es posible marcar/desmarcar el PFC como favorito para poder filtrar después por él, así como acceder a la información del tutor.



Figura 20 - Vista de tablón

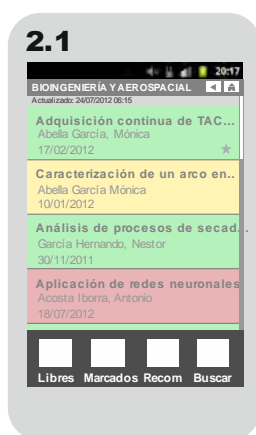


Figura 21 - Vista de listado de proyectos

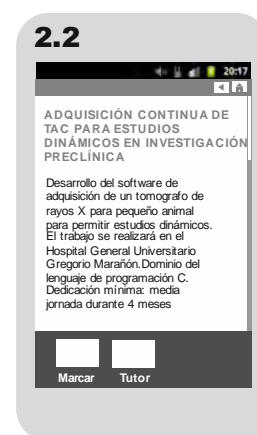


Figura 22 - Vista de detalle de proyecto

Tutor

Esta vista muestra los datos del tutor que dirige el proyecto.

Contacto

A través de la vista de tutor se puede acceder a un formulario de contacto. El

mecanismo de comunicación dependerá de los datos de contacto disponibles para el tutor y las posibilidades que ofrezca el SO.

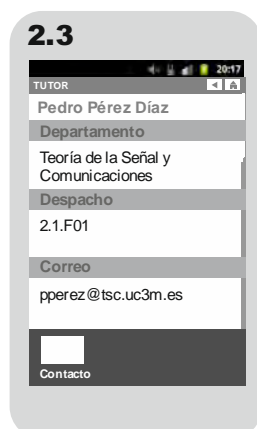


Figura 23 - Vista de tutor

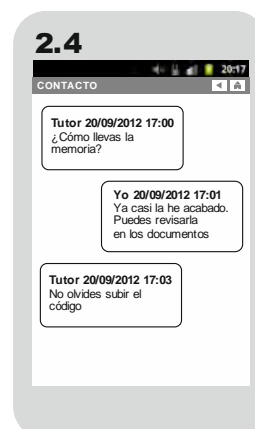


Figura 24 - Vista de contacto

Buscador

Permite la búsqueda avanzada de PFCs por los criterios definidos en el punto y que dependen de las prestaciones de la interfaz con el tablón de la UCM3. El botón de “Limpiar” permite resetear los campos a su valor por defecto, mientras que el botón de “Buscar” ejecuta el proceso de búsqueda y remite a la vista de listado de proyectos con los PFC que cumplen los criterios indicados.



Figura 25 - Vista de buscador

Información

Consiste en un listado con temas relacionados con los PFC. Seleccionando cada uno de ellos, el usuario es dirigido a la vista con el detalle de la información sobre ese tema.

Detalle de información

Contiene el detalle con la información sobre un tema seleccionado en la vista anterior.



Figura 26 - Vista de información



Figura 27 - Vista de detalle de información

Perfil

Esta vista permite al usuario añadir y editar sus datos personales. Estos datos quedan almacenados en la memoria del dispositivo y son utilizados por otras funcionalidades que los requieren, tales como el tablón o la matrícula.



Figura 28 - Vista de perfil

Proyecto

Contiene información sobre el proyecto matriculado. Es similar a la vista de detalle de PFC, pero incluirá información adicional sólo visible para un usuario matriculado.

Matricula

A través de esta vista el usuario podrá seleccionar y matricular un PFC. Está previsto su conexión con la vista de detalle de PFC para el acceso directo a la matriculación,

aunque en general su funcionalidad dependerá de las posibilidades que ofrezca la interfaz con Aula Global.

Seguimiento

A través de esta vista el usuario puede conocer el estado de avance de las tareas que implica un PFC. El objetivo es que el tutor pueda ir actualizando el porcentaje de avance según su estimación del trabajo de alumno.



Figura 29 - Vista de matrícula

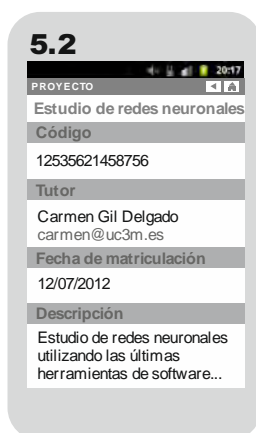


Figura 30 - Vista de proyecto



Figura 31 - Vista de seguimiento

Agenda

Esta vista cuenta con dos perspectivas: una a modo de listado y otra a modo de calendario, que pueden intercambiarse mediante un botón en la barra de herramientas inferior. En esta misma barra se encuentra el botón para añadir nuevos eventos.

Editar/añadir evento

Tanto para añadir como para editar un evento se emplea el mismo formulario, aunque en este último caso los contenidos del evento existente se muestran precargados.

Para crear un evento se indica el título, la ubicación y las fechas de inicio y fin. Se contempla la opción de añadir más campos, en función de las posibilidades que ofrezca la interfaz con la agenda/calendario del SO.



Figura 32 - Vista de agenda



Figura 33 - Vista de añadir/editar evento



Figura 34 - Vista de documentos

Documentos

Con el objetivo de centralizar la información del PFC se propone que durante su elaboración estudiante y tutor compartan documentos online. Esta vista facilitará la gestión de dichos documentos a través de las interfaces externas disponibles.

Ayuda

Contiene información sobre el manejo de las distintas funcionalidades de la aplicación.

Acerca de

En esta vista se muestra la información de versión y contacto de la aplicación.



Figura 35 - Vista de documentos

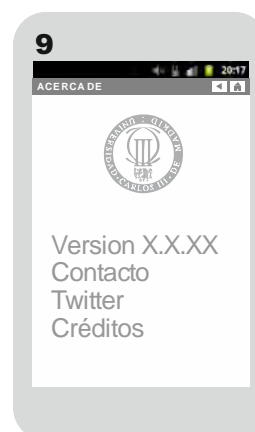


Figura 36 - Vista de acerca de

Diagrama de navegación

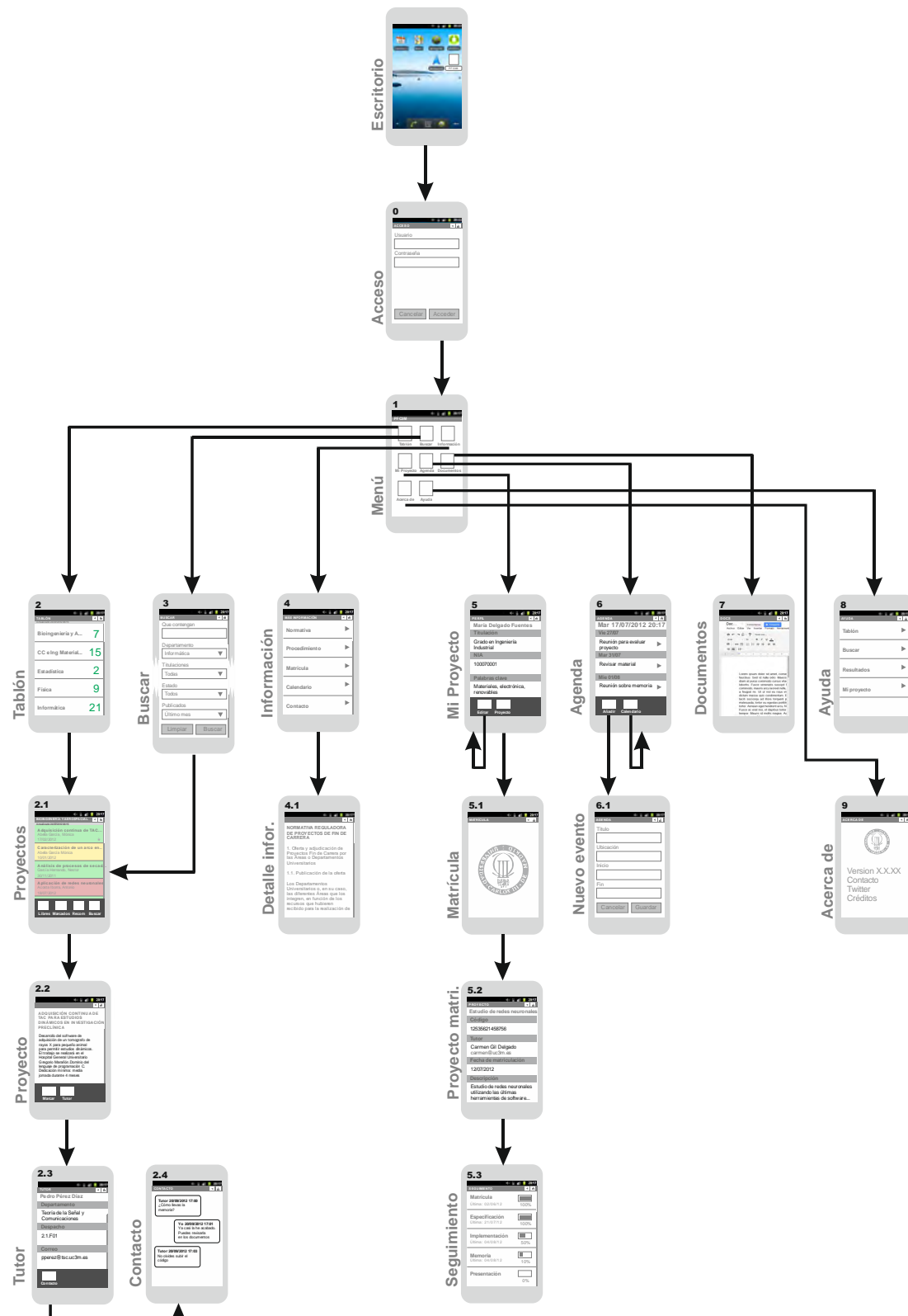


Figura 37 – Diagrama de navegación

4.2.2. Clases

En este apartado se describe el diseño del código Java, que equivale a la capa de *Control*. Según la función realizada, podemos distinguir cuatro conjuntos de clases o paquetes: Control de vistas, representación del modelo de datos, acceso a modelo remoto, acceso a modelo local. La Figura 38 ilustra la relación entre estos conjuntos de clases.

Control de vistas

Estas clases, representadas en Figura 39 y Figura 40, son las responsables de recoger y gestionar la interacción del usuario con las interfaces descritas en el apartado anterior. La interacción del usuario puede consistir en:

- Una consulta al modelo de datos de la aplicación (remoto o local).
- El acceso a una utilidad de la aplicación (como enviar un mensaje).
- Un cambio de vista (navegación).

Representación del modelo de datos

Estas clases representan entidades del modelo de datos (proyectos, departamentos...) para facilitar el acceso a sus campos. Algunas entidades complejas pueden disponer de un adaptador para facilitar la generación de las vistas. La Figura 41 muestra el diagrama UML (37) de estas clases.

Acceso a modelo remoto

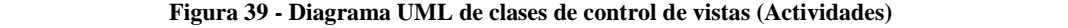
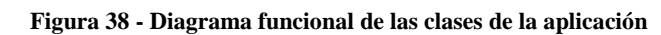
Estas clases (Figura 42) ofrecen un conjunto de servicios que transforman las peticiones que el usuario realiza a través las vistas en consultas al servicio web. Asimismo los servicios procesan la respuesta recibida y generan los objetos de entidades que los adaptadores requieren para generar las vistas.

Acceso a modelo local

A este conjunto corresponden las clases que proporcionan el acceso al modelo local de datos (preferencias de usuario, búsquedas recientes, favoritos...). En este caso, en lugar de peticiones al servidor se realizan consultas directas al sistema de almacenamiento seleccionado dentro del propio dispositivo. Como en el caso del acceso al modelo remoto, las clases transforman los datos en objetos de entidades. La Figura 43 muestra el diagrama UML de estas clases.

Utilidades

Por último existe un conjunto adicional de clases (Figura 44) que da soporte a funciones generales de la aplicación, tales como el formato de fechas.



Desarrollo de una aplicación para la gestión de PFCs/TFGs para dispositivos móviles Android

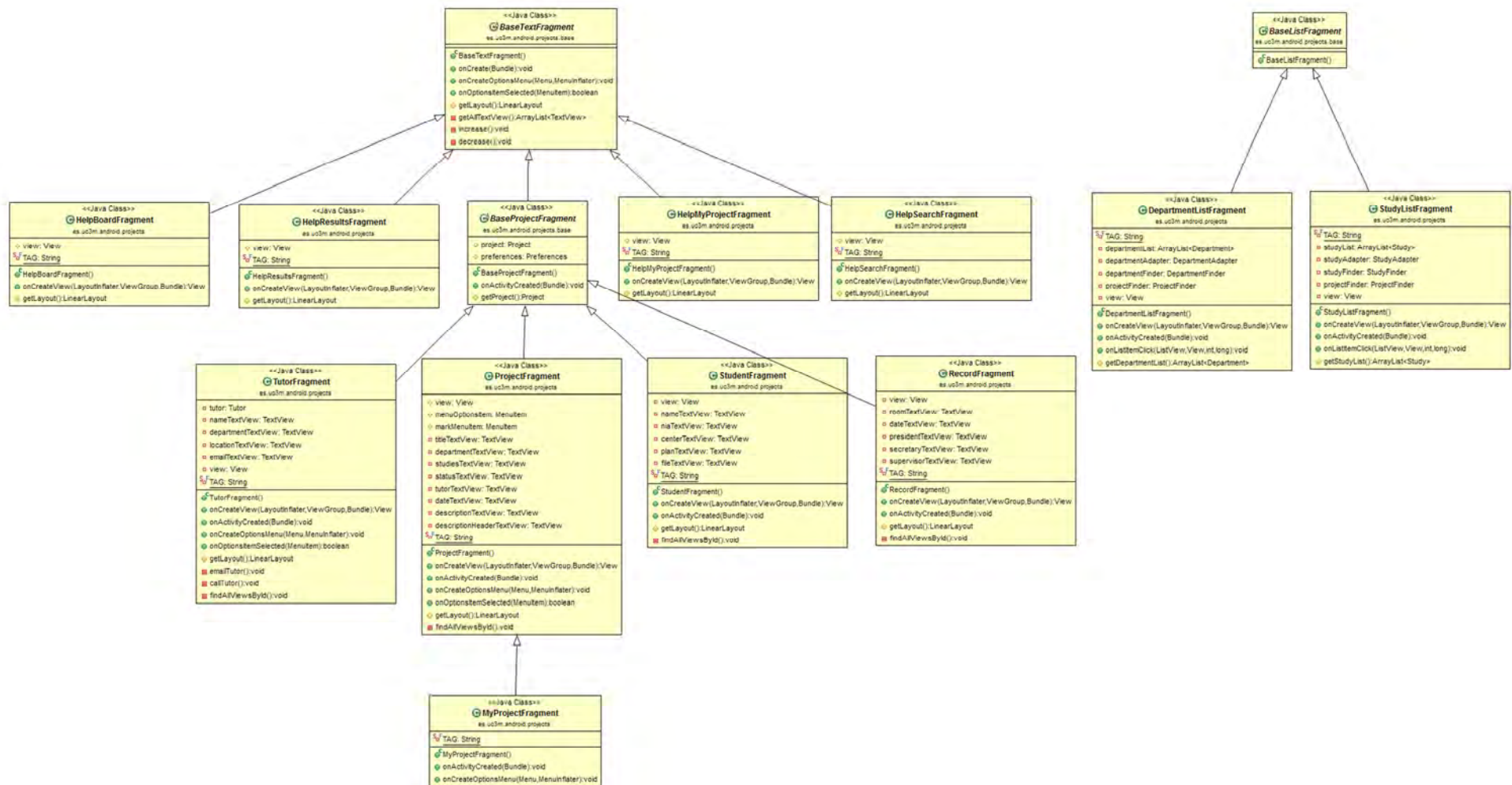


Figura 40 - Diagrama UML de clases de control de vistas (Fragmentos)

Desarrollo de una aplicación para la gestión de PFCs/TFGs para dispositivos móviles Android

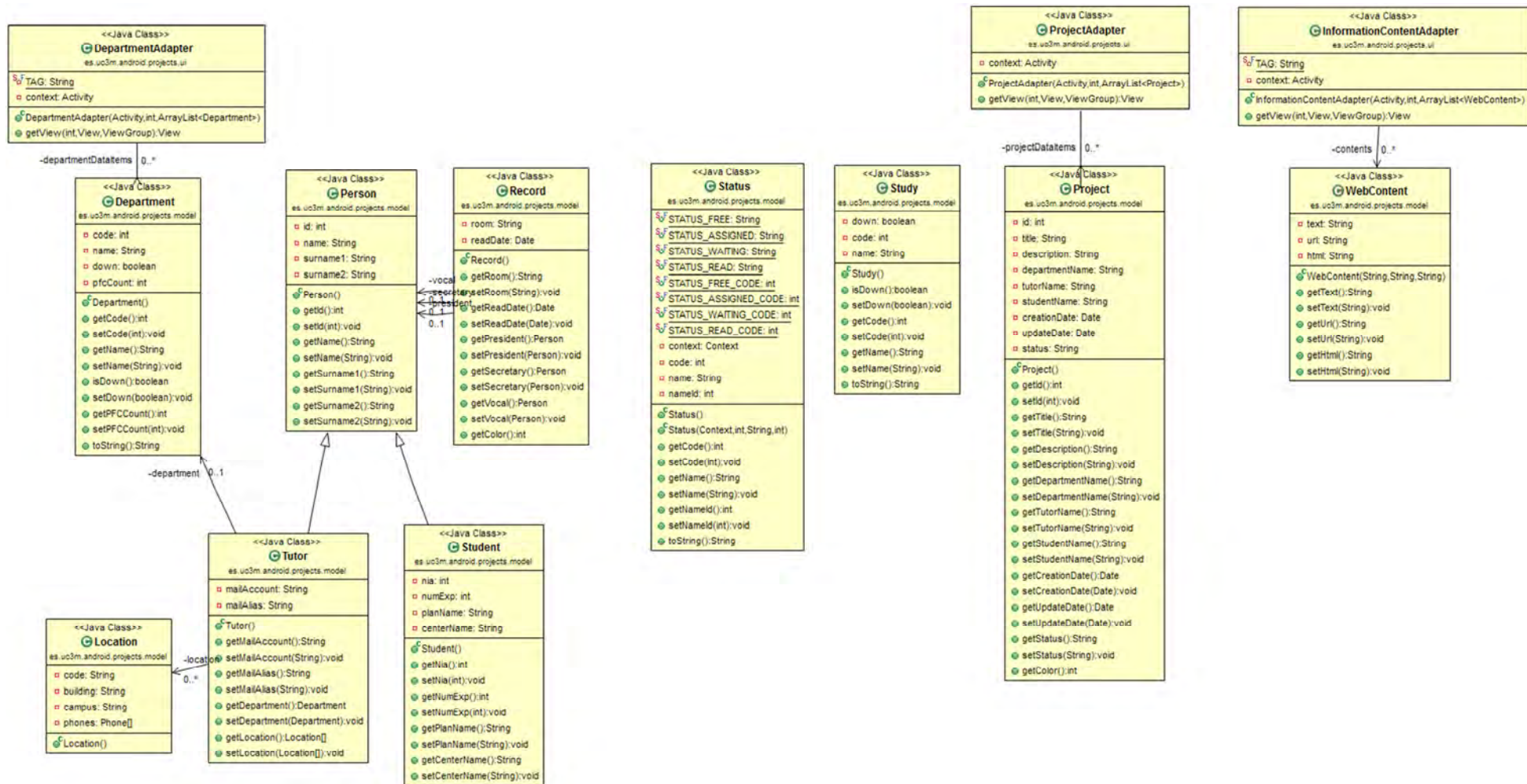


Figura 41 - Diagrama UML de clases entidades y adaptadores

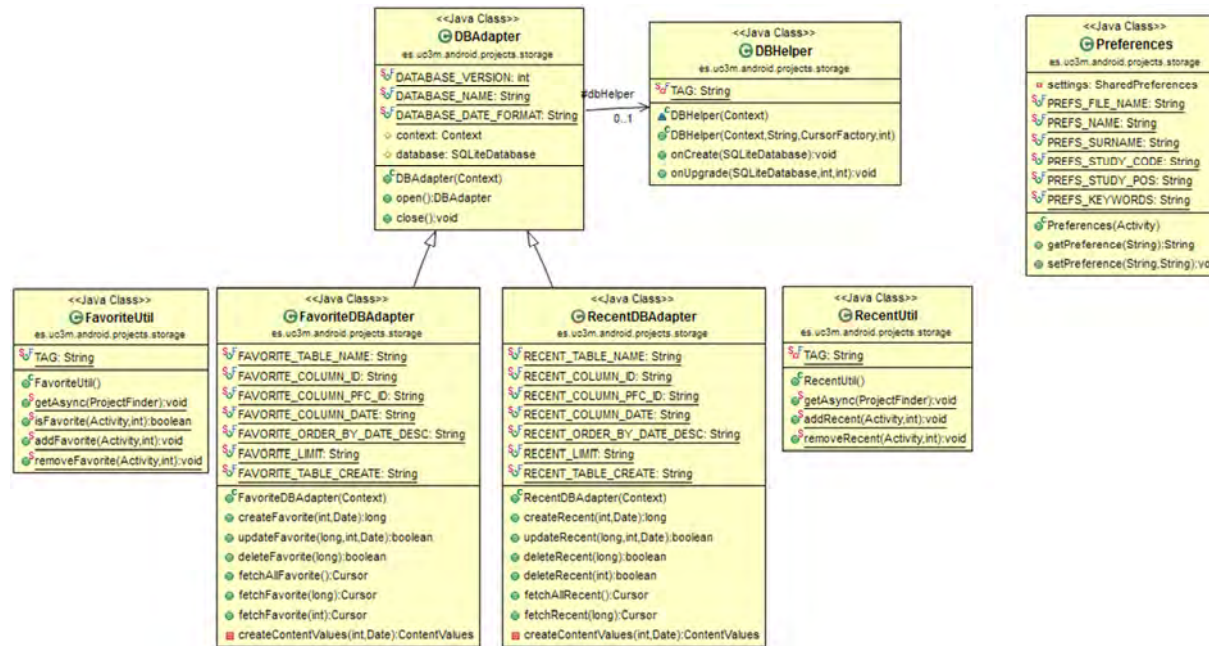


Figura 42 - Diagrama UML de clases de acceso al modelo local

Desarrollo de una aplicación para la gestión de PFCs/TFGs para dispositivos móviles Android

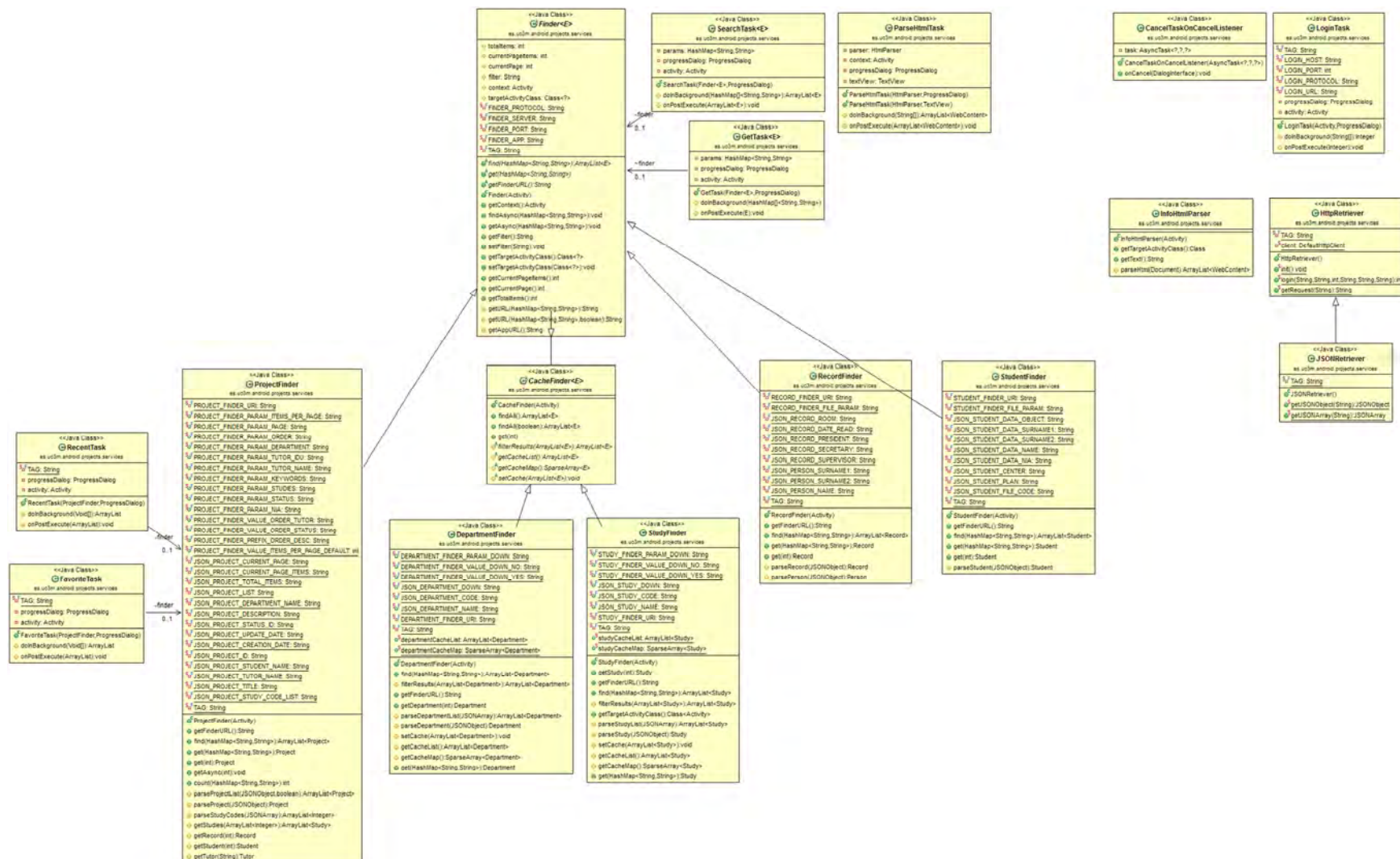


Figura 43 - Diagrama UML de clases de acceso al modelo remoto

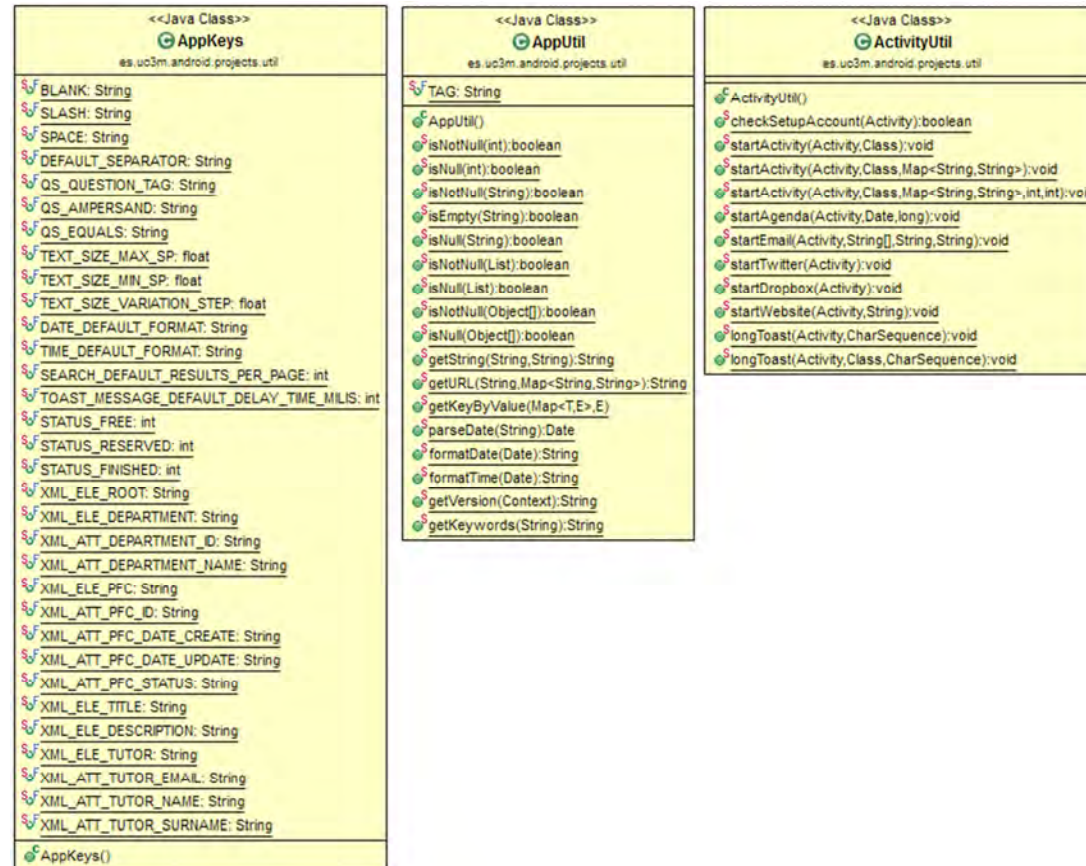


Figura 44 - Diagrama UML de clases de utilidad

4.2.3. Modelo de datos

En este apartado se describe el diseño del almacenamiento de datos *locales* de la aplicación, que corresponde a la capa de *Modelo*.

Proyectos recientes

Constituye un conjunto de registros de los proyectos visitados recientemente. Los campos de cada registro se describen en la tabla siguiente:

Campo	Descripción
Clave primaria	Identificador único autoincremental del registro
Id del proyecto	Identificador del proyecto consultado en el servicio web de la UC3M
Fecha	Fecha de la consulta

Tabla 47 – Registro de proyectos recientes

Proyectos favoritos

Constituye un conjunto de registros de los proyectos que el usuario ha marcado como favoritos. Los campos de cada registro se describen en la tabla siguiente:

Campo	Descripción
Clave primaria	Identificador único autoincremental del registro
Id del proyecto	Identificador del proyecto marcado en el servicio web de la UC3M
Fecha	Fecha en la que se añadió como favorito

Tabla 48 – Registro de proyectos favoritos

Perfil de usuario

En este caso se trata de un único registro con los datos del perfil del usuario de la aplicación. Los campos del registro se describen en la tabla siguiente:

Campo	Descripción
Nombre	Nombre del usuario
Apellidos	Apellidos del usuario
Id de la titulación	Identificador de la titulación del usuario en el servicio web de la UC3M
Palabras clave	Conjunto de hasta 5 palabras clave de interés para el usuario
Id del proyecto	Identificador del proyecto matriculado

Tabla 49 – Registro de perfil de usuario

4.3. Diseño de la aplicación servidora

La aplicación servidora ha sido creada expresamente por el Departamento de Servicios Informáticos de la UC3M para permitir el acceso remoto a datos de PFCs y TFGs. Esto supone una novedad respecto al resto de aplicaciones móviles de la UC3M, y pretende ser un modelo de trabajo para el futuro.

La especificación completa de la aplicación puede consultarse en la siguiente URL: http://aplicaciones.uc3m.es/pfc-api/apidocs/api_pfc_v1.html. A continuación se describirán de forma muy breve los servicios ofrecidos:

4.3.1. Búsqueda de proyectos

Este servicio ofrece las siguientes opciones:

- Obtener datos de un proyecto
- Obtener el expediente del alumno al que está asignado el proyecto
- Obtener el acta del proyecto si está leído
- Buscar proyectos mediante una serie de criterios

4.3.2. Listado de departamentos

Obtiene los departamentos existentes en la UC3M. A partir de esta información, pueden recuperarse los proyectos por cada departamento utilizando el servicio de búsqueda de proyectos y el id del departamento.

4.3.3. Listado de titulaciones

Obtiene las titulaciones ofertadas en la UC3M. A partir de esta información, pueden recuperarse los proyectos por cada titulación utilizando el servicio de búsqueda de proyectos y el id de la titulación.

4.3.4. Directorio

Se trata en realidad de una aplicación a parte, pero cuya funcionalidad resulta útil para la búsqueda de proyectos por tutor o para consultar la información de contacto del tutor.

5. Implementación

En este capítulo se explica el procedimiento seguido y los detalles de la implementación de la aplicación. Como referencia general para la programación de la aplicación Android se siguió el libro *Professional Android 4 Application Development* (38).

5.1. Modelo de datos interno de la aplicación

Como se describe en el apartado 4.2.3, la aplicación requiere el almacenamiento de una serie de datos en la memoria del dispositivo móvil. Todo el código necesario para la gestión del almacenamiento de datos se encuentra en el paquete `es.uc3m.android.projects.storage` de la aplicación.

Según las características de los datos a almacenar, se ha optado por dos mecanismos diferentes:

5.1.1. Preferencias compartidas

Las preferencias compartidas o *Shared Preferences* (39) permiten el almacenamiento persistente de pares clave-valor de tipos primitivos (booleanos, enteros, cadenas de texto...). Esta ha sido la opción elegida para los datos del perfil del usuario recogidos en la Tabla 49 – Registro de perfil de usuario.

El almacenamiento y recuperación de las preferencias se realiza a través de la clase `Preferences`.

5.1.2. Base de datos

Android ofrece soporte completo para bases de datos *SQLite* (40). Esta ha sido la opción elegida para almacenar los datos indicados en Tabla 47 – Registro de proyectos recientes y Tabla 48 – Registro de proyectos favoritos.

Siguiendo las recomendaciones de Android, la clase `DBHelper` se encarga de la creación, actualización y borrado de las tablas. Para cada entidad (en este caso, Recientes y Favoritos) existe una extensión de la clase `DBAdapter` para crear las consultas a la base de datos. Finalmente, para permitir un acceso sencillo a las funciones CRUD de estas entidades, cada una dispone de una clase `Util` que abstrae la creación y cierre de conexiones con la base de datos.

5.2. Integración con el servicio web JSON

La mayor parte del modelo de datos de la aplicación reside en el servidor de datos de proyectos. Este servidor expone una API de servicios web JSON que permite el acceso a dichos datos.

Este desarrollo es uno de los más complejos de la aplicación. El código generado se encuentra localizado en los paquetes `es.uc3m.android.projects.services` y `es.uc3m.android.projects.model` de la aplicación.

Para facilitar la reutilización del código y la extensión de los servicios en un futuro, se ha implementado una clase abstracta y genérica denominada *Finder*, que define la lógica necesaria para el acceso a los servicios, el procesamiento de la respuesta JSON y la generación de objetos de modelo o POJOs (41) para facilitar su uso por las capas superiores de la aplicación.

A continuación se describe con más detalle la implementación de cada una de estas funcionalidades.

5.2.1. Autenticación

El acceso a los servicios web de la UC3M requiere la autenticación del usuario con sus credenciales de Aula Global. Cuando se inicia la aplicación, el usuario debe introducir dichas credenciales en la vista de acceso. La aplicación realiza entonces una petición de autenticación mediante el cliente HTTP (42) de la clase *HttpRetriever*. Si las credenciales de acceso son válidas, el cliente HTTP las mantendrá internamente en futuras peticiones de modo que el usuario no deberá introducirlas de nuevo. **En ningún momento se almacena esta información en el sistema de ficheros del dispositivo.**

5.2.2. Petición a los servicios web

La documentación de los servicios web de proyectos de la UC3M describe en detalle las URLs y parámetros necesarios para la realización de consultas. Para cada servicio, su extensión correspondiente de la clase *Finder* construye la URL necesaria a partir de los parámetros recibidos y realiza la consulta mediante la clase *JSONRetriever*.

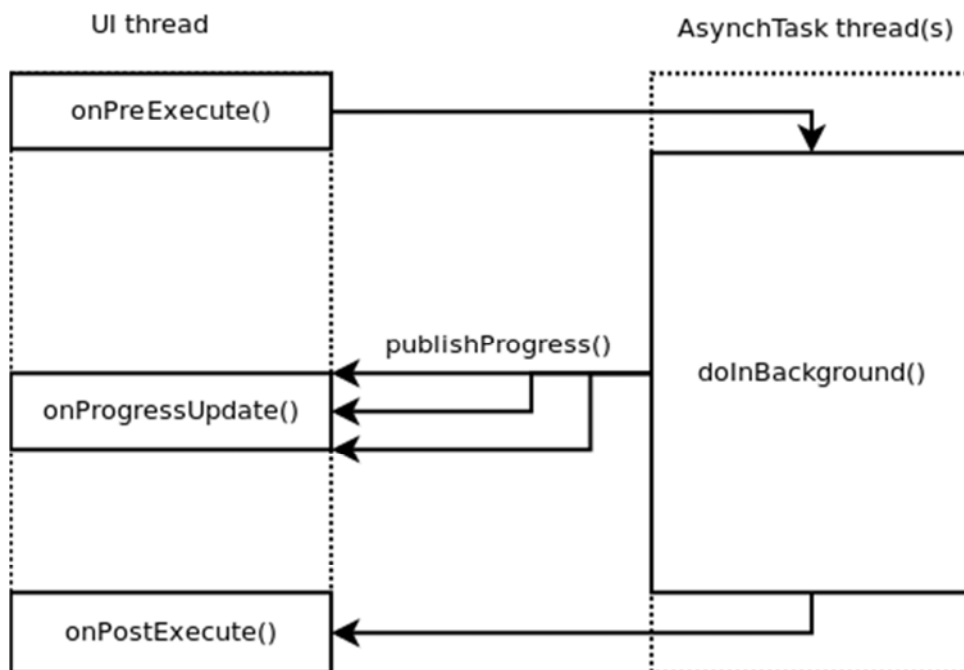


Figura 45 - Modelo de ejecución asíncrono de tareas (43)

El acceso a los servicios web implica realizar una tarea de duración indeterminada.

Para que la ejecución de dicha tarea no bloquee completamente la aplicación hasta su finalización, Android ofrece un mecanismo para lanzar procesos de forma asíncrona (44). La clase `Finder` utiliza las clases `*Task` del paquete para realizar consultas a los servicios de forma asíncrona. Mientras, en el hilo principal (o hilo UI) se muestra un mensaje que informa al usuario de que hay una tarea en proceso.

De este modo, el procesamiento de la respuesta y la generación de objetos se lleva a cabo como una tarea “de fondo” (`doInBackground`). Cuando estas tareas finalizan (`onPostExecute`), se abre la vista correspondiente con los datos obtenidos.

5.2.3. Objetos de modelo

Para facilitar la generación de las vistas de la aplicación, toda la información recuperada a partir de los servicios web debe exponerse de forma sencilla. Por este motivo los métodos de búsqueda de las clases `*Finder` no retornan directamente el resultado de la consulta o los objetos JSON, sino objetos o listas de objetos básicos Java (POJOs).

Estos objetos, contenidos en el paquete `es.uc3m.android.projects.model`, tienen las siguientes características comunes:

- Extienden la clase `Model`, que a su vez implementa la interfaz `Serializable`. Esto es necesario para poder pasar estos objetos como parte de los datos de una vista.
- Sólo contienen campos de tipos básicos o serializables, y sus correspondientes métodos de acceso. En algunos casos pueden contener alguna lógica adicional.
- Sobrescriben el método `toString` para facilitar tanto su uso en las vistas como el tratamiento de errores y la depuración del código.

5.2.4. Caché de objetos

Por la naturaleza de algunos servicios (consultas repetitivas, número de resultados dentro de un rango limitado) resulta conveniente almacenar los datos obtenidos como objetos en memoria. De esta forma se mejora notablemente el rendimiento de la aplicación, reduciendo el número de peticiones y los tiempos de respuesta.

Para ello se ha utilizado una extensión de la clase `Finder` denominada `CacheFinder`. Este tipo de buscador se ha utilizado para los servicios de búsqueda de departamentos y de titulaciones. Cuando se realiza el primer inicio de la aplicación (o cualquier inicio después del cierre completo de la misma), la vista `SplashActivity` utiliza la clase `StartupTask` para poblar la caché de estos objetos, de forma que estén disponibles durante toda la sesión sin necesidad de realizar más consultas.

5.3. Obtención de datos de páginas web

Como parte de los requisitos de la aplicación, el usuario debe poder consultar la

información relevante sobre la redacción y defensa de los proyectos de fin de carrera y fin de grado. Actualmente no existe ningún servicio web o repositorio de documentación donde se proporcionen estos datos.

La dirección web http://www.uc3m.es/portal/page/portal/biblioteca/aprende_usar/TFG contiene enlaces a varios documentos con información relevante sobre TFGs y PFCs. De forma experimental se ha optado por procesar y recuperar de los enlaces a partir del HTML de esta página. Esta solución tiene la ventaja de que si se añaden, actualizan o eliminan enlaces, la aplicación lo reflejará de forma automática sin tener que realizar cambios; pero tiene la desventaja de que un cambio en el marcado o diseño de la página requeriría ajustar el código de la aplicación.

La clase `es.uc3m.android.projects.services.HtmlParser` utiliza la librería de código abierto *JSOUP* (45) para procesar y obtener los enlaces con información a partir de la URL indicada. Como ocurre con el acceso a los servicios web, la tarea se lleva a cabo de forma asíncrona utilizando la clase `ParseHTMLTask`.

The screenshot shows the UC3M Library website. At the top, there's a header with the university logo and name. Below it, the word 'Biblioteca' is prominently displayed. A navigation bar includes links like 'Inicio', 'La Universidad', 'Servicios Universitarios', 'Biblioteca', 'Aprende a usar tu Biblioteca', and 'Trabajo de Fin de Grado'. A search bar is also present. The main content area is titled 'Trabajo de Fin de Grado' and includes a sub-header '1. Reunir información'. This section is divided into two columns: '1.1. Encontrar información' and '1.2. Evaluar la información'. The '1.1' column lists types of projects (scientific, technical, humanistic, artistic, design, and business) and mentions a 'quién es el autor' section. The '1.2' column describes a critical analysis process. On the right side, a sidebar titled 'Más información' is highlighted with a red box, containing links to 'Normativa oficial para el TFG', 'E-archivo: Trabajos Académicos (TFG, TFM y PFC)', 'E-archivo: Proyectos de Fin de Carrera Escuela Politécnica Superior', 'Modalidad de Trabajo de Fin de Grado "Emprende"', and several 'El TFG en Secretaría Virtual' links for different faculties.

Figura 46 – Enlaces con información de PFCs y TFGs

5.4. Implementación de las interfaces gráficas de usuario

En este apartado se describe la implementación de la interfaz gráfica de usuario (GUI) de la aplicación. En general, la interfaz gráfica consiste en un conjunto de *Actividades* (46) enlazadas entre sí mediante las opciones de navegación, tal y como se describe en el apartado 4.2.1.

5.4.1. Diseño

Aunque inicialmente se optó por mantener un diseño similar al de las aplicaciones existentes de la UC3M (Ver apartado 2.2), finalmente se optó por seguir las líneas de diseño actuales recomendadas por Android (47). Entre otras cosas, se recomienda el uso de colores e iconos planos.



Figura 47 – Versiones inicial y final del estilo de la aplicación

Otra de las diferencias principales radica en el uso de la barra de acciones (48) para alojar los iconos y menús de la aplicación. Debido a que la barra de acciones sólo está soportada desde versiones 3.0 (API 11) en adelante y que uno de los requisitos de la aplicación es la compatibilidad con versiones 2.3 (Ver apartado 3.4.2) fue necesario utilizar una librería denominada *ActionBarSherlock* (49) que emula el comportamiento de la barra de acciones en versiones 2.x de Android.

El tema de estilo de la aplicación (denominado *Uc3m*) está a su vez basado en el estilo *Holo Light* con barra de acción oscura, adaptado al uso de *ActionBarSherlock*. Para general el tema, se utilizó la herramienta online de creación de widgets y temas *Android Asset Studio* (50).

Los recursos gráficos se han organizado siguiendo el patrón estándar para aplicaciones Android (51). De esta forma, bajo la carpeta */res* encontramos:

- *drawable*: contiene los elementos gráficos. En algunos casos, se proporciona

un recurso diferente según la resolución de la pantalla (-ldpi, -mdpi, -hdpi, -xhdpi).

- `layout`: contiene los ficheros `.xml` que definen la estructura de las vistas de la aplicación.
- `menu`: contiene los ficheros `.xml` que definen la estructura de los menús de la aplicación.
- `values`: contiene a su vez:
 - `color`: definiciones de colores
 - `styles`: definiciones de estilos. Existe una variante para versiones 14 y superiores de iOS.
 - `strings`: pares clave-valor para las traducciones. En la versión actual sólo se soporta inglés (estándar) y español (-es).

5.4.2. Navegación

Como parte de las decisiones de diseño tomadas durante la implementación, se optó por utilizar una navegación basada en menú deslizante en lugar del menú en rejilla utilizado por algunas aplicaciones de la UC3M. El principal motivo es facilitar el acceso a las vistas de la aplicación con una sola acción, sin tener que regresar siempre al menú.

Para implementar esta funcionalidad se utilizó el proyecto *Facebook Like Slideout Navigation* diseñado por Alex Korovyansky y compartido en *Github* (52). En la aplicación, las clases utilizadas para la creación del menú se encuentran en el paquete `es.uc3m.android.projects.menu`.

Finalmente, se ha optado por utilizar pestañas o *tabs* en la navegación interna de cada sección.

5.4.3. Vistas

Para la implementación de las vistas de la GUI se ha primado la reusabilidad del código. Por este motivo, en el paquete `es.uc3m.android.projects.base` se han añadido una serie de *Actividades* y *Fragments* básicos que contienen funcionalidades comunes tales como el acceso a la navegación. Todas las vistas de la aplicación extienden de una clase de este paquete y se encuentran localizadas en el paquete raíz de la aplicación `es.uc3m.android.projects`.

Listados

En el caso del tablón, los listados de departamentos y titulaciones son extensiones de la clase `BaseListFragment`, ya que se muestran como tabs de la vista principal `BoardActivity`. En cambio, los listados de proyectos `ProjectListActivity` e

información `InformationActivity` extienden de la clase `BaseListActivity`.

Como se explicó en el apartado 4.2.1, los listados requieren de un adaptador entre el objeto de tipo modelo obtenido desde los servicios y el layout diseñado para un elemento de la lista. Estos adaptadores se encuentran en el paquete `es.uc3m.android.projects.ui`.

Detalles

Las vistas de detalle son *Fragments* (53) que se utilizan para mostrar la información de una entidad del modelo de datos (proyecto, tutor, estudiante...). Los fragmentos pueden reutilizarse en distintas actividades, como ocurre en el caso de `ProjectActivity` (detalle de un proyecto) y `MyProjectActivity` (vista de Mi Proyecto).

Cada fragmento de detalle agrega al menú de la barra de acciones sus propias opciones. Las opciones comunes, definidas en `BaseTextFragment`, son las de incrementar y reducir el tamaño de la letra para cumplir con el requisito de accesibilidad. Por este mismo requisito, todas las opciones del menú disponen de un atajo de teclado.

Formularios

Los formularios son un tipo de vista en la que el usuario introduce información para obtener un resultado. En la aplicación existen dos formularios: `SearchActivity` (buscador) y `ProfileFragment` (perfil, como parte de Mi Proyecto).

5.5. Integración con otras aplicaciones

Para cumplir algunos requisitos del proyecto es necesario hacer uso de otras aplicaciones instaladas en el dispositivo. Android ofrece dos aproximaciones para este caso:

- Definir la acción que se desea realizar y permitir al usuario elegir la aplicación con la que ejecutarla. Por ejemplo, en el caso de enviar un mensaje la aplicación sólo especifica la acción `ACTION_SEND`, de forma que el usuario puede elegir entre cualquiera de las aplicaciones instaladas que pueda llevar a cabo esta acción.
- Abrir una aplicación específica. En el caso de compartir documentos, la aplicación intenta abrir *Dropbox* como aplicación del dispositivo. Si no estuviese disponible, muestra la versión web.

La clase `es.uc3m.android.projects.util.ActivityUtil` contiene, entre otros, una serie de métodos para abrir las aplicaciones necesarias.

6. Pruebas

6.1. Introducción

En este capítulo se detallan las pruebas realizadas para valorar el cumplimiento de los requisitos marcados durante la fase de análisis.

6.2. Entornos de prueba

A continuación se detallan las características de los entornos utilizados para las pruebas de la aplicación:

6.2.1. Emulador (versión 2.3)

Para probar la compatibilidad de la aplicación con la versión 2.3 se utilizó el emulador incluido en el entorno de desarrollo de Android (Android Virtual Device). La configuración del emulador se detalla en la tabla siguiente:

Característica	Valor
Tamaño de pantalla	3.2''
Resolución	QVGA (320x480) mdpi
Versión de la aplicación	2.3.3 (API 10)
Memoria física	512 MB
Almacenamiento	250MB

Tabla 50 – Características técnicas del emulador para las pruebas

6.2.2. Entorno real (versión 4.2)

Para las pruebas en entorno real se utilizó un dispositivo Android *Samsung Galaxy Note 10.1* con las siguientes características técnicas (54):

Característica	Valor
Tamaño de pantalla	10.1''
Resolución	WXGA (1280x800) xhdpi
Versión de la aplicación	4.2 (API 17)
Memoria física	2 GB
Almacenamiento	16GB
Procesador	Exynos 4 núcleos de 1,4GHz
Conexión	Wifi

Tabla 51 – Características técnicas del emulador para las pruebas

También se realizaron pruebas en un dispositivo Samsung Galaxy S3 proporcionado por la tutora del proyecto.

6.3. Cumplimiento de los requisitos de usuario

En este apartado se evalúa el cumplimiento de los requisitos fijados en los apartados 3.3. y 3.4.

6.3.1. Requisitos de capacidad

Evaluación

La tabla siguiente refleja el resultado de la evaluación de los requisitos de capacidad y sus correspondientes requisitos de software funcionales con la versión entregada de la aplicación:

RUC	RSF	Título	Cumplimiento
01	01,02,03	Buscar proyectos	Completo
02	01,02,03	Filtrar los resultados obtenidos	Completo
03	04	Acceder a los detalles de un proyecto	Completo
04	05	Contactar con el tutor de un proyecto	Completo
05	07	Crear un perfil de usuario	Completo
06	03	Encontrar los proyectos que mejor se ajustan al perfil	Completo
07	06	Acceder a la información oficial sobre proyectos	Completo
08	09	Recuperar búsquedas recientes	Completo
09	04,08	Acceder a la información del proyecto matriculado	Parcial
10	10	Administrar eventos del calendario sobre proyectos	Completo
11	12	Acceder a la ayuda de la aplicación	Completo
12	05,11	Compartir documentos con el tutor del proyecto	Completo
13	09	Marcar proyectos como favoritos	Completo
14	11	Compartir la información del proyecto	Completo
15	13	Matricular un proyecto	Ninguno
16	11	Informar al tutor de la evolución del proyecto	Ninguno

Tabla 52 – Evaluación del cumplimiento de los requisitos de capacidad

Observaciones

El grado de cumplimiento de los requisitos de capacidad de la aplicación es alto, aunque como se refleja en la tabla no se han podido completar parcial o totalmente algunas funcionalidades:

- Acceso a la información del proyecto matriculado: la API de servicios web no proporciona actualmente esta posibilidad. Como alternativa se permite al usuario marcar manualmente un proyecto como seleccionado.
- Matricular un proyecto: la API de servicios web no proporciona esta opción
- Informar al tutor de la evolución del proyecto: Se ha descartado porque requeriría de una herramienta de backoffice para tutores que permitiese recoger la información enviada por el estudiante.

6.3.2. Requisitos de restricción

Evaluación

La tabla siguiente refleja el resultado de la evaluación de los requisitos de restricción y sus correspondientes requisitos de software no funcionales con la versión entregada de la aplicación:

RUR	RSNF	Título	Cumplimiento
01	01	Compatibilidad	Completo
02	02	Acceso a información remota	Completo
03	02,03	Tiempos de espera	Parcial
04	03	Rendimiento	Completo
05	04	Protección de datos	Completo
05	05	Extensibilidad	Completo
06	06	Idioma de la aplicación	Completo
07	07	Accesibilidad	Completo

Tabla 53 – Evaluación del cumplimiento de los requisitos de restricción

Observaciones

El grado de cumplimiento de los requisitos de restricción de la aplicación es alto, aunque como se refleja en la tabla para los tiempos de espera no se ha logrado el resultado esperado. La recuperación de departamentos y titulaciones con su número de proyectos es costosa, ya que requiere dos peticiones por cada departamento/titulación. Esto provoca que el tiempo de primera carga de la aplicación sea mayor de lo deseado.

7. Planificación y presupuesto

En este anexo se detallan la planificación y el presupuesto de la realización de este proyecto.

7.1. Planificación

La tabla siguiente recoge el número de días (jornadas de 8 horas) requerido en una de las tareas necesarias para llevar a cabo el proyecto:

Tarea	Duración (días)
Propuesta y Especificación de Requisitos del Software	5
Estudio del estado del arte	10
Análisis del sistema	15
Diseño del sistema	20
Implementación	50
Pruebas	10
Reuniones de seguimiento tutor/Sistemas Informáticos	5
Desarrollo de la aplicación servidora	50 (Estimado)
Memoria	20
TOTAL	185

Tabla 54 – Planificación del proyecto

7.2. Presupuesto

En este apartado se detalla el presupuesto necesario para el desarrollo de este proyecto, teniendo en cuenta los costes de personal, hardware y software.

7.2.1. Costes de personal

En la tabla siguiente se calculan los costes de personal, teniendo en cuenta un coste medio de 100€/jornada incluyendo las retenciones de IRPF y Seguridad Social.

Concepto	Coste jornada (€)	Jornadas	Coste (€)
Jornadas ingeniero (app. móvil)	100	130	13.000
Jornadas ingeniero (app. servidora)	100	50	5.000
Jornadas asesoría tutor/SI	100	5	500
TOTAL	100	185	18.500

Tabla 55 – Costes de personal

7.2.2. Costes de hardware

En la tabla siguiente se calculan los costes del hardware necesario para desarrollar el proyecto (Aplicación cliente):

Concepto	Coste Unidad (€)	Unidades	Coste (€)
Samsung Galaxy Note 10.1	449	1	450
HP Probook 4520s	699	1	699
TOTAL	1.148	2	1.148

Tabla 56 – Costes de hardware

7.2.3. Costes de software

En la tabla siguiente se calculan los costes del software necesario para desarrollar el proyecto (Aplicación cliente):

Concepto	Coste Unidad (€)	Unidades	Coste (€)
Eclipse + ADK	0	1	0
Microsoft Office 2010	120	1	120
Licencia de desarrollo Android*	25	1	25
TOTAL	125	2	125

Tabla 57 – Costes de hardware

*Sólo si se pretende publicar la aplicación en Google Play

7.2.4. Coste total

Concepto	Coste (€)
Personal	13.100
Hardware	1.148
Software	125
TOTAL SIN IMPUESTOS	19.773
Impuestos (21%)	4.152,33
TOTAL	23.925,33

Tabla 58 – Coste total del proyecto

8. Conclusiones

En este capítulo se extraen las conclusiones obtenidas tras la realización del proyecto. Asimismo se proponen una serie de líneas futuras con el objetivo de fomentar la mejora de la aplicación en próximos proyectos.

8.1. Conclusiones

8.1.1. Aspectos positivos

La aplicación entregada satisface en gran medida los objetivos marcados al inicio del proyecto, ya que permite a los alumnos y profesores la búsqueda avanzada de PFCs y TFGs y una primera aproximación a la gestión de los mismos. Es en este segundo objetivo donde se debe mejorar más en futuras versiones.

Respecto a los requisitos, como se describe en el Capítulo 6 en general la aplicación ha cumplido con todos los marcados como prioritarios..

Por otra parte, el diseño del código de la aplicación permitirá no sólo su extensión en el futuro sino su reutilización parcial para la implementación de otras aplicaciones basadas en servicios web JSON de la UC3M.

Finalmente, la experiencia obtenida en el diseño y desarrollo de la aplicación podrá ser utilizada por este alumno en su carrera profesional, al tratarse de un conocimiento muy demandado actualmente.

8.1.2. Aspectos negativos

Aunque en general la API de servicios web para proyectos cubre las funcionalidades más importantes, al no disponer de opciones de matriculación o gestión más avanzadas no se han podido cumplir algunos de los objetivos de la aplicación.

En cuanto al diseño de la aplicación, al no disponer de una guía o plantilla de aplicaciones de la UC3M, es posible que algunas de las decisiones no se correspondan exactamente con los criterios de diseño y programación de las aplicaciones de la Universidad.

8.2. Líneas futuras

A continuación se proponen algunas líneas futuras para la mejora de la aplicación:

8.2.1. Mejoras en la aplicación

- Definición de tareas y seguimiento. A través de una aplicación *backoffice* el tutor podría crear en un servidor las tareas que el usuario debe completar como parte de su proyecto. La aplicación recuperaría la información de las tareas mediante

un servicio web y permitiría al alumno actualizar el avance en cada una de ellas. Esta información sería enviada al tutor para que mediante la aplicación *backoffice* evalúe la evolución de las tareas.

- Reserva y matriculación de proyectos. Si los servicios web proporcionan la opción de reservar y matricular un proyecto, la aplicación debería incorporar estas opciones.
- Versión optimizada para tablets Android. Como ocurre con la aplicación UC3M en iOS, la aplicación de proyectos podría disponer de una versión optimizada para dispositivos Android de mayores dimensiones como tabletas.
- Versiones para otros sistemas operativos. La aplicación podría tomarse como base para crear aplicaciones equivalentes para otros sistemas operativos como iOS o Windows Mobile.

8.2.2. *Mejoras en los servicios web*

- Mejora de los servicios de departamentos y titulaciones. La respuesta de estos servicios debería proporcionar el número de proyectos disponibles en cada uno. De esta forma el procesamiento de estos datos sería más eficiente al ahorrar peticiones al servicio web de proyectos.
- Mejoras en el servicio de proyectos. Los datos de proyectos deberían incluir las palabras clave que los tutores añaden para etiquetarlos, así como cualquier otra información que sea de utilidad a la hora de buscar proyectos.
- Servicio de documentación. Los enlaces a la documentación existente sobre PFCs y TFGs debería ser proporcionados mediante un servicio web con lo que no sería necesario extraer los datos del código HTML de la página web.
- Servicio para reserva y matriculación de proyectos: Los servicios web deberían permitir realizar operaciones seguras para usuarios tales como reservar y matricular un proyecto.

Referencias

1. Eclipse. [En línea] [Citado el: 10 de 10 de 2012.] <http://www.eclipse.org/>.
2. iTunes. [En línea] [Citado el: 1 de Febrero de 2013.] <https://itunes.apple.com>.
3. Google Play. [En línea] [Citado el: 1 de Febrero de 2013.] <https://play.google.com/store/apps>.
4. Oferta de proyectos Fin de Carrera y Trabajos Fin de Grado. [En línea] [Citado el: 28 de Enero de 2013.] <http://www3.uc3m.es/web/pfc/index.xml>.
5. Wikipedia. *BlackBerry OS*. [En línea] [Citado el: 11 de Enero de 2013.] http://es.wikipedia.org/wiki/BlackBerry_OS.
6. **López, Rubén G.** IT Espresso. [En línea] 2010. [Citado el: 11 de Enero de 2013.] <http://www.itespresso.es/la-mensajeria-pin-a-pin-de-blackberry-45655.html>.
7. **Honan, Mathew.** Macworld. *Apple unveils iPhone*. [En línea] [Citado el: 11 de Enero de 2013.] <http://www.macworld.com/article/1054769/iphone.html>.
8. Unix Network Scientific Computing. [En línea] [Citado el: 20 de Mayo de 2013.] <http://unixnetwork.wordpress.com/2013/02/18/darwin-bsd-2/>.
9. Windows Phone. *What's new in Windows Phone 8*. [En línea] [Citado el: 11 de Enero de 2013.] <http://www.windowsphone.com/en-us/how-to/wp8/start/whats-new-in-windows-phone>.
10. **Valero, Claudio.** ADSL Zone. [En línea] 2012. [Citado el: 12 de Enero de 2013.] <http://www.adslzone.net/article9843-android-alcanza-en-numero-de-aplicaciones-a-ios.html>.
11. **Wikipedia.** Mobile Operating System. *Market Share*. [En línea] [Citado el: 20 de Mayo de 2013.] http://en.wikipedia.org/wiki/Mobile_operating_system#Market_share.
12. Soporte Blackberry. [En línea] [Citado el: 11 de Enero de 2013.] <http://developer.blackberry.com>.
13. Soporte Apple. [En línea] [Citado el: 11 de Enero de 2013.] <http://developer.apple.com>.
14. Soporte Windows Phone. [En línea] [Citado el: 11 de Enero de 2013.] <http://developer.windowsphone.com>.

15. Soporte Android. [En línea] [Citado el: 11 de Enero de 2013.] <http://developer.android.com>.
16. **Android, Xataka.** [En línea] [Citado el: 20 de Mayo de 2013.] <http://www.xatakandroid.com/mercado/ya-hay-mas-de-750-millones-de-dispositivos-android-en-el-mundo>.
17. Xatakandroid. *El 28,4% de los dispositivos Android llevan Jelly Bean.* [En línea] 2013. [Citado el: 20 de Mayo de 2013.] <http://www.xatakandroid.com/mercado/el-28-4-de-los-dispositivos-android-llevan-jelly-bean>.
18. GNU Operating System. *GNU General Public License, version 2.* [En línea] [Citado el: 15 de Febrero de 2013.] <http://www.gnu.org/licenses/gpl-2.0.html>.
19. The Apache Software Foundation. *Apache License, Version 2.0.* [En línea] [Citado el: 20 de Mayo de 2013.] <http://www.apache.org/licenses/LICENSE-2.0.html>.
20. Arquitectura Android. [En línea] [Citado el: 13 de Abril de 2013.] <https://sites.google.com/site/swcuc3m/home/android/generalidades/2-2-arquitectura-de-android>.
21. App Brain. *Number of available Android applications.* [En línea] [Citado el: 31 de 10 de 2013.] <http://www.appbrain.com/stats/number-of-android-apps>.
22. W3C Working Group Note. *Web Service Glossary.* [En línea] 2004. [Citado el: 20 de Mayo de 2013.] <http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/#webservice>.
23. **C., Benjamín González.** Desarrollo Web. *SOAP (Simple Object Access Protocol).* [En línea] 2004. [Citado el: 21 de Abril de 2013.] <http://www.desarrolloweb.com/articulos/1557.php>.
24. Architectural Styles and the Design of Network-based Software Architectures. **Fielding, Roy Thomas.** s.l. : UNIVERSITY OF CALIFORNIA, IRVINE, 2000.
25. Java Community Process. *JSR 311: JAX-RS: The Java™ API for RESTful Web Services.* [En línea] [Citado el: 12 de Marzo de 2013.] <http://jcp.org/en/jsr/detail?id=311>.
26. W3C. *Extensible Markup Language (XML).* [En línea] 2003. [Citado el: 20 de Mayo de 2013.] <http://www.w3.org/XML/>.
27. JSON. *Introducción a JSON.* [En línea] [Citado el: 1 de Marzo de 2013.] <http://www.json.org/json-es.html>.
28. **Riecken, Jim.** Edugarage. *AJAX and JSON - Lessons Learned.* [En línea] [Citado el: 11 de Enero de 2013.]

- http://www.edugarage.com/download/attachments/42369042/24020_AJAX.JSON_Riecken.pdf.
29. **Locken, Shandra.** Aurora EDI Alliance. *The What, Why and How of JSON for EDI Integration Specialists*. [En línea] 2013. [Citado el: 15 de Septiembre de 2013.] <http://www.auroraedialliance.com/>.
 30. JSON. *JSON in JavaScript*. [En línea] [Citado el: 1 de Marzo de 2013.] <http://www.json.org/js.html>.
 31. Ley Orgánica 15/1999 de 13 de diciembre, de Protección de Datos de Carácter Personal. *Boletín Oficial del Estado*. [En línea] 14 de diciembre de 1999. <http://www.boe.es/boe/dias/1999/12/14/pdfs/A43088-43099.pdf>.
 32. **devices, Opinion 02/2013 on apps on smart.** WP29 Opinion on apps on mobile devices. *European Commission*. [En línea] Marzo de 2013. http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2013/wp202_en.pdf.
 33. **López, Juan José Marín.** Gomez-Acebo & Pombo. [En línea] Junio de 2013. <http://www.gomezacebo-pombo.com/media/k2/attachments/la-proteccion-de-datos-en-las-aplicaciones-de-los-dispositivos-inteligentes.pdf>.
 34. **Android.** Security Tips. *Android Developers*. [En línea] [Citado el: 20 de Octubre de 2013.] <http://developer.android.com/training/articles/security-tips.html>.
 35. Wikimedia Commons. *Client-Server Model*. [En línea] 2011. [Citado el: 12 de 10 de 2013.] <http://upload.wikimedia.org/wikipedia/commons/c/c9/Client-server-model.svg>.
 36. **University, Department of Computer Science. Central Connecticut State.** Model-View-Controller pattern. [En línea] [Citado el: 13 de Enero de 2013.] <http://www.cs.ccsu.edu/~stan/classes/CS355/notes/03-AndroidUI.html>.
 37. Object Management Group. *Unified Modeling Language*. [En línea] 2013. <http://www.omg.org/spec/UML/>.
 38. **Meier, Reto.** Professional Android 4 Application Development. 2012. ISBN: 978-1-118-10227-5.
 39. Shared Preferences. [En línea] [Citado el: 20 de Abril de 2013.] <http://developer.android.com/guide/topics/data/data-storage.html#pref>.
 40. SQLite en Android. [En línea] [Citado el: 2 de Mayo de 2013.] <http://developer.android.com/guide/topics/data/data-storage.html#db>.
 41. Plain Old Java Objects. [En línea] [Citado el: 1 de Marzo de 2013.] <http://www.martinfowler.com/bliki/POJO.html>.

42. Apache. *HTTP Components*. [En línea] [Citado el: 12 de 9 de 2013.] <http://hc.apache.org/httpclient-3.x/authentication.html>.
43. Ejecución asíncrona de tareas. [En línea] [Citado el: 3 de Mayo de 2013.] <http://developer.att.com/developer/forward.jsp?passedItemId=11900168>.
44. Procesos e hilos en aplicaciones Android. [En línea] [Citado el: 11 de Mayo de 2013.] <http://developer.android.com/guide/components/processes-and-threads.html>.
45. JSOUP HTML parser. [En línea] [Citado el: 20 de Abril de 2013.] <http://jsoup.org/>.
46. Actividades en Android. [En línea] [Citado el: 11 de Mayo de 2013.] <http://developer.android.com/guide/components/activities.html>.
47. Design Guidelines for Android Applications. [En línea] [Citado el: 4 de Abril de 2013.] <http://developer.android.com/design/index.html>.
48. Action Bar. [En línea] [Citado el: 4 de Mayo de 2013.] <http://developer.android.com/guide/topics/ui/actionbar.html>.
49. ActionBarSherlock. [En línea] [Citado el: 4 de Mayo de 2013.] <http://actionbarsherlock.com/>.
50. Android Asset Studio. [En línea] [Citado el: 12 de Abril de 2013.] <http://android-ui-utils.googlecode.com/hg/asset-studio/dist/index.html>.
51. Recursos gráficos en Android. [En línea] [Citado el: 12 de Mayo de 2013.] <http://developer.android.com/guide/topics/resources/providing-resources.html>.
52. **Korovyansky, Alex**. Android Facebook-like Slideout Navigation. [En línea] [Citado el: 20 de Abril de 2013.] <https://github.com/korovyansk/android-fb-like-slideout-navigation>.
53. Fragmentos en Android. [En línea] [Citado el: 3 de Mayo de 2013.] <http://developer.android.com/guide/components/fragments.html>.
54. Samsung Galaxy Note 10.1 - Especificaciones técnicas. [En línea] [Citado el: 20 de Mayo de 2013.] http://www.samsung.com/es/galaxynote_10.1/specifications.html.

ANEXO A. Guía de usuario

Este anexo contiene la guía de usuario de la aplicación desarrollada.

A.1. Iniciar la aplicación

El inicio de la aplicación se realiza mediante el icono disponible en el escritorio o en el administrador de aplicaciones del dispositivo.



Figura 48 – Guía de usuario: icono de la aplicación

Para poder acceder a la aplicación el usuario debe aceptar la política de protección de datos, introducir sus credenciales de Aula Global (usuario y contraseña) y pulsar el botón de Acceder.



Figura 49 – Guía de usuario: acceso a la aplicación

Cuando se accede a la aplicación por primera vez (o después de haberla finalizado completamente) se muestra la vista de inicio. En este momento se realiza la carga de algunos datos de la aplicación, por lo que pueden transcurrir unos segundos hasta que la vista se cierra y se muestra el tablón.



Figura 50 – Guía de usuario: pantalla inicial

A.2. Navegar por la aplicación

El icono de menú ubicado a la izquierda en la barra de acciones para todas las vistas permite navegar rápidamente por la aplicación.



Figura 51 – Guía de usuario: menú de navegación

A.3. Buscar proyectos

La aplicación ofrece dos formas de localizar proyectos: seleccionando el departamento o la titulación en que se ofrecen en el tablón, o bien realizando una búsqueda avanzada.

Tablón de departamentos y titulaciones

La vista contiene dos pestañas, Departamentos y Titulaciones, las cuales contienen un listado con los departamentos y titulaciones que ofrecen proyectos, respectivamente, y el número de proyectos ofrecidos.



Figura 52 – Guía de usuario: tablón proyectos por departamentos y titulaciones

Utilizando el cuadro de texto superior, se pueden filtrar los departamentos o titulaciones por nombre.

A través del icono de búsqueda situado a la derecha en la barra de acciones se accede al buscador de proyectos.

Buscador

Además de localizar proyectos a través de su departamento o titulación en el tablón, se puede utilizar el buscador para encontrar proyectos en base a otros criterios.

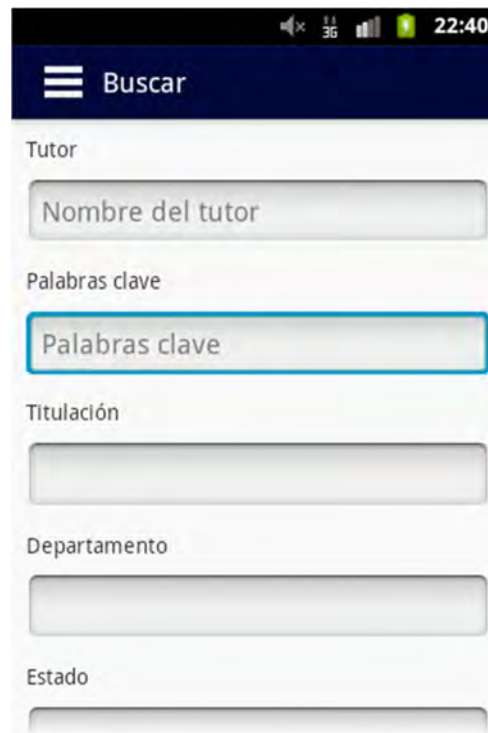


Figura 53 – Guía de usuario: buscador

Es necesario rellenar al menos uno de los criterios de búsqueda disponibles:

- Tutor: nombre del tutor responsable del proyecto.
- Palabras clave: hasta cinco palabras clave contenidas en el título o la descripción del proyecto.
- Titulación: titulación en la que se oferta el proyecto, a elegir del listado.
- Departamento: departamento en el que se oferta el proyecto, a elegir del listado.
- Estado: estado del proyecto, a elegir del listado.



Figura 54 – Guía de usuario: selección de departamento en el buscador



Figura 55 – Guía de usuario: selección de palabras clave en el buscador

Una vez definidos los criterios es necesario pulsar el botón “Buscar” para iniciar la búsqueda.

A.4. Acceder a los proyectos encontrados

Listado de proyectos

Cuando se realiza una búsqueda de proyectos, bien sea desde el tablón o desde el buscador, los resultados se muestran en forma de listado. Los proyectos aparecen coloreados según el estado en que se encuentran:

- Verde: el proyecto está libre
- Amarillo: el proyecto ha sido asignado
- Blanco: el proyecto está pendiente de lectura
- Rojo: el proyecto ha sido leído.

Si se han encontrado más de 20 resultados, cuando se llega al final del listado aparece un botón para mostrar los 20 siguientes.

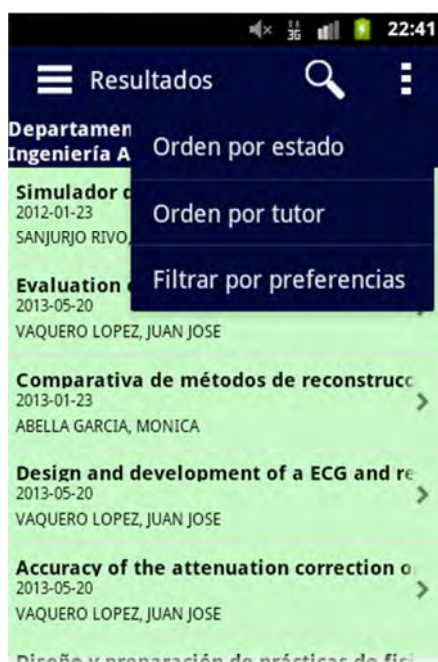


Figura 56 – Guía de usuario: listado de proyectos

Además del acceso directo al buscador, el menú del listado de proyectos ofrece las opciones siguientes:

- Ordenar el listado por el estado del proyecto
- Ordenar el listado por el nombre del tutor del proyecto
- Filtrar el listado por las preferencias fijadas en el perfil de usuario (ver A.5)

Detalle de proyecto

Al pulsar sobre un resultado de búsqueda se muestran los detalles del proyecto seleccionado.



Figura 57 – Guía de usuario: detalles de proyecto

El tamaño de la letra puede ajustarse mediante los iconos correspondientes de la barra de aplicaciones. Además, se ofrecen las siguientes opciones:

- Marcar proyecto como favorito para ser consultado más tarde desde el perfil del usuario.
- Marcar el proyecto como proyecto del usuario, de forma que a partir de ese momento su información aparezca en la vista de *Mi Proyecto*.
- Compartir la información del proyecto a través de correo o de las redes sociales.

La pestaña de tutor contiene los detalles del profesor responsable del proyecto. Asimismo, desde el menú el usuario puede ponerse en contacto con el tutor por teléfono o mediante correo electrónico.

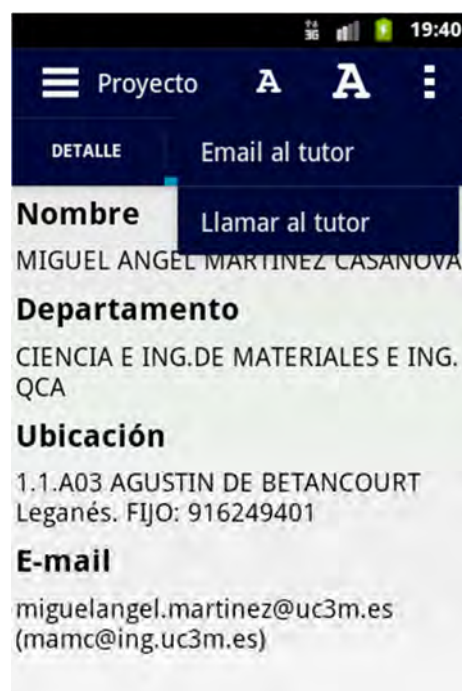


Figura 58 – Guía de usuario: tutor del proyecto

Si el proyecto ya ha sido asignado o está pendiente de lectura, la pestaña de alumno mostrará los datos del estudiante que lo tiene asignado.



Figura 59 – Guía de usuario: alumno del proyecto

Finalmente, si el proyecto ha sido leído, la pestaña *acta* muestra la información del acta de lectura del proyecto.



Figura 60 – Guía de usuario: acta del proyecto

A.5. Definir un perfil de usuario

Definir un perfil permite al usuario personalizar los criterios de búsqueda. Para ello hay que seleccionar la opción *Mi proyecto* desde el menú de navegación y rellenar los campos del formulario presentado en la pestaña *Perfil*.



Figura 61 – Guía de usuario: mi proyecto

Desde el menú de opciones de esta vista, el usuario puede acceder a los proyectos marcados como favoritos, consultar los proyectos vistos recientemente o realizar una búsqueda en base a las preferencias configuradas en su perfil.

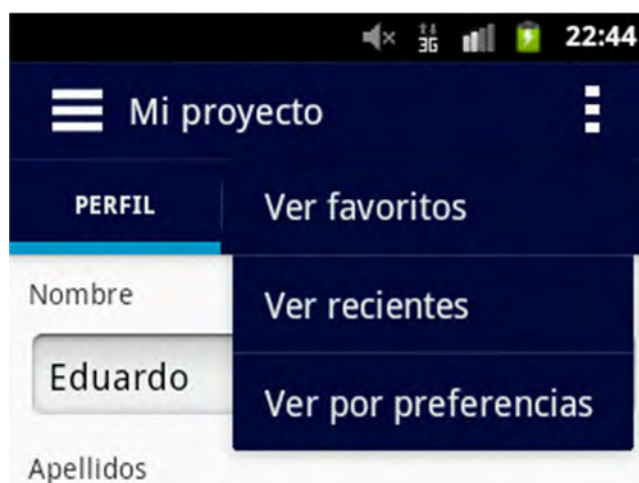


Figura 62 – Guía de usuario: opciones del perfil de usuario

Si el usuario ha marcado un proyecto como suyo, en la vista se muestran dos pestañas adicionales con los detalles del proyecto y del tutor.

A.6. Obtener información sobre proyectos

Seleccionando la opción *Información* en el menú de navegación se accede a la documentación sobre PFCs y TFGs disponible online.



Figura 63 – Guía de usuario: información sobre proyectos

A.7. Gestionar los eventos de la agenda

La opción *Agenda* del menú de navegación pueden agregarse eventos relacionados con la búsqueda, redacción y presentación de TFGs y PFCs.

A.8. Compartir documentos

A través de la opción *Documentos* del menú se accede a la aplicación *Dropox*. Si esta aplicación no está instalada en el dispositivo, se muestra su versión web.

A.9. Ver la ayuda

La sección *Ayuda* muestra varias pestañas con información sobre el funcionamiento de distintos aspectos de la aplicación.



Figura 64 – Guía de usuario: ayuda

A.10. Obtener información sobre la aplicación

Finalmente, la sección *Acerca de* contiene información sobre la versión de la aplicación y permite contactar con sus responsables mediante correo o *Twitter*.



Figura 65 – Guía de usuario: acerca de



Figura 66 – Guía de usuario: créditos

ANEXO B. Guía de implantación

En este anexo se describen los pasos para instalar la aplicación.

B.1. Requisitos mínimos

A continuación se enumeran los requisitos mínimos para la instalación de la aplicación:

- Dispositivo Android con versión 2.3 o superior
- Conexión a internet
- Aplicación Google Play Store instalada (para instalación desde Google Play)
- Cable USB u otro tipo de conexión con el sistema de ficheros del dispositivo

B.2. Instalación desde Google Play Store

Los pasos para instalar la aplicación a través de la aplicación Google Play Store se detallan a continuación:

1. Iniciar la aplicación Google Play Store en el dispositivo Android donde se desea instalar la aplicación.
2. Localizar la aplicación a través del buscador.
3. Abrir la página de detalles y seleccionar la opción *Descargar*.
4. Pulsar *Aceptar* y *descargar* para aceptar los permisos de la aplicación.

B.3. Instalación a partir del APK

Los pasos para instalar la aplicación mediante el fichero .apk se enumeran a continuación:

1. En el dispositivo donde se desea instalar la aplicación seleccionar *Ajustes* → *Aplicaciones* y habilitar la opción *Orígenes desconocidos*.
2. Transferir el fichero .apk de la aplicación al dispositivo mediante una conexión USB, una tarjeta SD o cualquier otro mecanismo.
3. Seleccionar el fichero en el dispositivo y seleccionar *Instalar*.
4. Aceptar los permisos requeridos para finalizar la instalación.

ANEXO C. Material entregado

En este apartado se enumeran los contenidos del CD entregado para la evaluación de este Trabajo de Fin de Grado.

C.1. Instalable

En la carpeta **/instalable** se incluye el fichero **projectsuc3m.apk** para la instalación de la aplicación en un dispositivo Android compatible siguiendo los pasos descritos en el apartado B.3.

C.2. Código fuente de la aplicación

En la carpeta **/codigo** se incluye todo el código fuente de la aplicación

C.3. Memoria

La carpeta **/memoria** contiene el documento de la memoria de la aplicación en formatos .docx y .pdf.